Defence Research and Development Canada | Recherche et développement pour la défense Canada

DEFENCE R&D DÉFENSE

Automation of IED Threat Emplacement for Training Scenarios

Final Report

David Unrau, Richard Zobarich, Catherine Levoir
CAE PS Canada

Prepared By:
CAE PS Canada Inc.
1135 Innovation Drive
Ottawa, Ontario
K2K 3G7
Consulting Services
Contractor's Document Number: 5180-001 ver 05
Contract Project Manager: Richard Percival, 613-247-0342
PWGSC Contract Number: W7711-06-8100-14
CSA: Dr. Jerzy Jarmasz, Defence Scientist, 416-635-2000

Defence R&D Canada
Contract Report
DRDC Toronto CR 2011-134
March 2012

Canada

Automation of IED Threat Emplacement for Training Scenarios

Final Report

David Unrau, Richard Zobarich, Catherine Levoir
CAE PS Canada

Prepared By:
CAE PS Canada Inc.
1135 Innovation Drive
Ottawa, Ontario
K2K 3G7
Consulting Services
Contractor's Document Number: 5180-001 ver 05
Contract Project Manager: Richard Percival, 613-247-0342
PWGSC Contract Number: W7711-06-8100-14
CSA: Dr. Jerzy Jarmasz, Defence Scientist, 416-635-2000

Defence R&D Canada – Toronto

Contract Report
DRDC Toronto CR 2011-134
March 2012

Principal Author

Original signed by David Unrau

David Unrau

Senior Consultant

Approved by

Original signed by Dr. Jerzy Jarmasz

Dr. Jerzy Jarmasz

Defence Scientist, Learning and Training Group

Approved for release by

Original signed by Dr. Stergios Stergiopoulos

Dr. Stergios Stergiopoulos

Acting Chair, Knowledge and Information Management Committee, Acting Chief Scientist

Abstract

In contemporary operations, asymmetric threats, especially Improvised Explosive Devices (IEDs), are a leading cause of Canadian Forces casualties and injuries. Automation to support the development of better threat scenarios for training exercises would improve the Canadian Forces' capability to prepare soldiers for future missions involving emerging threats. This report summarizes progress to date on the design of a software tool to automate the generation of plausible IED threat scenarios. The core components of the design are a knowledge processing engine and a geospatial processing engine. The knowledge processing engine will act on a database (ontology) of insurgent tactics to translate user-supplied constrains and training objectives into threat scenario characteristics. The geospatial processing engine will query map data to determine locations for scenario components, such as the device location and spotter locations. Initial technical prototyping has demonstrated the feasibility of this approach. An iterative operator-machine interface development process, cycling through design, prototyping and evaluation phases is suggested as the next step in development.

Résumé

Dans les opérations contemporaines, les menaces asymétriques, en particulier les dispositifs explosifs de circonstance (IED), sont une cause majeure de blessures et de décès chez le personnel des Forces canadiennes. L'automatisation servant à appuyer le développement de meilleurs scénarios de menace pour les exercices de formation améliorerait la capacité des Forces canadiennes à préparer les soldats pour des missions à venir mettant en cause de nouvelles menaces. Ce rapport résume le progrès réalisé jusqu'à maintenant sur la conception d'un outil logiciel servant à automatiser la production de scénarios de menace d'IED plausibles. Les éléments essentiels du concept sont un moteur de traitement des connaissances et un moteur de traitement géospatial. Le moteur de traitement des connaissances agira sur une base de données (ontologie) de tactiques des insurgés pour traduire des objectifs de formation et des contraintes fournies par l'utilisateur en caractéristiques de scénario de menaces. Le moteur de traitement géospatial interroge les données cartographiques pour déterminer les emplacements pour les éléments scénario, comme l'emplacement du dispositif et les emplacements des guetteurs. Le prototypage technique initial a démontré la faisabilité de cette approche. Un processus de développement d'interface opérateur machine de type itératif, parcourant les phases de conception, de prototypage et d'évaluation est suggéré comme prochaine étape en développement.

This page intentionally left blank.

Executive summary

Automation of IED Threat Emplacement for Training Scenarios

David Unrau; Richard Zobarich; Catherine Levoir; DRDC Toronto CR 2011-134; Defence R&D Canada – Toronto; March 2012.

Introduction: In contemporary operations, asymmetric threats, especially Improvised Explosive Devices (IEDs), are a leading cause of Canadian Forces (CF) casualties and injuries. The CF must prepare for and adapt to insurgent tactics, as insurgent forces seem to adapt to patterns of behaviour set by coalition troops. Defence Research and Development Canada (DRDC) has established a Counter-IED technology demonstration project to address this issue.

A limitation encountered in Counter-IED training is the ability to produce IED scenarios that are structured "...in accordance with a good understanding of insurgent tactics and constraints imposed by the terrain itself..." Developing a method to automate the development of realistic threat scenarios would improve the CF's capability to prepare soldiers for future missions involving emerging threats. This report summarizes progress to date on the design of a software tool to automate the generation of plausible IED threat scenarios. Available literature on IED attacks and insurgent tactics has been reviewed. An initial cognitive task analysis has been performed, producing a decomposition and definition of insurgent behaviour in relation to IED attacks. This information has been used to develop the architecture and preliminary design of an automated IED threat scenario generation tool.

Results: A software architecture has been developed. A knowledge processing engine acts on an ontology of insurgent tactics to translate user-supplied constraints and training objectives into threat scenario characteristics. A geospatial processing engine queries map data to produce locations for IED attack scenario components, such as the device and spotter locations. Scenario information will be saved in standard and open forms, and could be exported to specific training systems, including synthetic environments.

Initial prototyping has demonstrated the feasibility of this approach. An initial ontology has been developed, and deduction of plausible attack configurations has been demonstrated to a limited extent. Key geospatial processing tools specified during the requirements analysis process have been prototyped.

Significance: By modeling possible IED attack scenarios based on data about real insurgent activity, this work takes the first steps in automating the process of developing Counter-IED training scenarios for simulation-based training in the CF. This work could also serve as a basis for analytic tools in support of Counter-IED mission planning and intelligence efforts.

Future plans: An iterative operator-machine interface development process is suggested as the next step in development. Three iterations are suggested, progressing from user evaluation of the high level concept through evaluation of a preliminary design to evaluation of a prototype system. Each iteration would be composed of sub-iterations through design, prototyping and evaluation activities.

Sommaire

Automatisation d'emplacement de menace de dispositif explosif de circonstance pour scénarios de formation

David Unrau; Richard Zobarich; Catherine Levoir ; DRDC Toronto CR 2011-134 ; R & D pour la défense Canada – Toronto; octobre 2011.

Introduction ou contexte : Dans les opérations contemporaines, les menaces asymétriques, en particulier les dispositifs explosifs de circonstance (IED), sont une cause majeure de blessures et de décès chez le personnel des Forces canadiennes (FC). Les FC doivent se préparer aux tactiques des insurgés et s'y adapter, car les forces révolutionnaires semblent s'adapter aux modèles de comportement établis par les troupes de la Coalition. Recherche et développement pour la défense Canada (RDDC) a mis sur pied un projet de démonstration de technologie anti IED pour s'attaquer à cette question. Une limite à laquelle on se bute dans la formation anti IED est la capacité à produire des scénarios IED qui sont structurés conformément à une bonne compréhension des contraintes et tactiques des insurgés imposées par le terrain lui même (y compris les aspects culturels comme la présence de populations locales ou de repères importants). La mise au point d'une méthode servant à automatiser le développement de scénarios de menace réalistes améliorerait la capacité des FC à préparer les soldats pour de futures missions mettant en cause de nouvelles menaces.

Ce rapport résume le progrès réalisé jusqu'à maintenant sur la conception d'un outil logiciel servant à automatiser la production de scénarios de menace d'IED plausibles. La documentation disponible sur les attaques par IED et les tactiques des insurgés a été revue. Une analyse de tâche cognitive initiale a été effectuée, produisant une décomposition et une définition du comportement de révolte en lien avec les attaques par IED. Cette information a été utilisée pour développer l'architecture et la conception préliminaire d'un outil de production de scénario de menaces par IED de type automatisé.

Résultats : Les éléments essentiels de la conception sont un moteur de traitement des connaissances et un moteur de traitement géospatial. Le moteur de traitement des connaissances agira sur une base de données (ontologie) de tactiques de révolte pour traduire des objectifs de formation et des contraintes fournies par l'utilisateur en caractéristiques de scénario de menaces. Le moteur de traitement géospatial interroge les données cartographiques pour produire les emplacements pour les éléments scénario d'attaque par IED, comme l'emplacement du dispositif et les emplacements des guetteurs. Les renseignements sur le scénarios seront sauvegardés dans des formulaires standard et ouverts, et pourraient être exportés vers des systèmes de formation spécifiques, y compris les environnements synthétiques. Le prototypage initial a démontré la faisabilité de cette approche. Une ontologie initiale a été créée, et la déduction de configurations d'attaque de type plausible a été démontrée dans une certaine mesure. Les outils de traitement géospatiaux clés spécifiés pendant le processus d'analyse des exigences ont été prototypés.

Importance : En modélisant les scénarios d'attaque par IED possibles en fonction des données concernant l'activité révolutionnaire réelle, ce travail entame l'automatisation du processus de développement de scénarios de formation anti IED pour la formation basée sur la simulation au

sein des FC. Ce travail pourrait aussi servir de base aux outils analytiques pour appuyer les efforts en matière de renseignements et de planification de missions anti IED.

Perspectives : Un processus de développement d'interface opérateur machine de type itératif est suggéré comme prochaine étape en développement. Trois itérations sont suggérées, allant de l'évaluation par l'utilisateur du concept de haut niveau jusqu'à l'évaluation d'un système prototype en passant par l'évaluation d'une concept préliminaire. Chaque itération serait composée de sous itérations par le biais d'activités de conception, de prototypage et d'évaluation.

This page intentionally left blank.

Table of contents

List of figures

List of tables

This page intentionally left blank.

1 Introduction

This document is the final report for task 14 under the Human Behaviour Representation (HBR) standing offer: *Automation of threat emplacement for training scenarios in synthetic environments*.

1.1 Background

In contemporary operations, asymmetric threats, especially Improvised Explosive Devices (IEDs), are a leading cause of Canadian Forces (CF) casualties and injuries. The CF must prepare for and adapt to insurgent tactics, as insurgent forces seem to adapt to patterns of behaviour set by coalition troops. Defence R&D Canada (DRDC) has established a counter-IED technology demonstration program [6] to address this issue:

> As a result of the increased threat to Canadian Forces (CF) personnel from IEDs, Defence Research and Development Canada (DRDC) has stood up a Technology Demonstration Program (TDP) on counter-IED technologies (the C-IED TDP, project code 12rr), aiming to deliver useable technologies to front-line users in a short time span to help mitigate the IED threat. This large research effort includes a number of sub-projects working to improve a range of soldier systems. One of these projects, the IED Awareness Training project (12rr03) led out of DRDC Toronto, is working to develop training systems for the CF in order to better prepare soldiers to make the best use of their current systems to detect IEDs and assess the level and nature of IED threat in their environment during convoy operations.

As discussed above, the IED Awareness Training project seeks to produce training systems to better prepare soldiers for convoy operations in areas of insurgent threat. A number of initiatives under this project simulate IED scenarios, or present information about insurgent tactics. A common limitation is the ability to produce IED scenarios that are structured "...in accordance with a good understanding of insurgent tactics and constraints imposed by the terrain itself (including cultural aspects such as the presence of local populations or significant landmarks..." [6] Developing a method to automate the development of realistic threat scenarios would improve the CF's capability to prepare soldiers for future missions involving emerging threats.

1.2 Scope

This task is the first step in the development of an automated emplaced threat scenario generation tool for simulation and training. The over-arching objectives for this tool are:

1. Ease-of-use for the end-user,
2. Configurable from the user as to what type of attack, the complexity of the attack, the skill of the insurgents, and the training objectives,
3. Adaptable to a wide variety of geographic locations and compatible with GIS data that is available to the CF,

4. Provides advice to the end-user on the suitability of a scenario from the current body of knowledge of insurgent activities,

5. Easy to update with respect to the CF's lessons learned from operations in IED environments, and

6. Enables the re-use of threat scenarios, ideally across simulation environments.

With the above list as the guiding objectives, the specific scope of this task was to:

1. Document the current state of CF knowledge about insurgent IED attack tactics,

2. Develop tools that apply knowledge of IED tactics to geographic datasets to produce components of plausible IED scenarios, and

3. Further develop the architecture and design of an end-user tool that supports the objectives listed above.

This report details the progress to date on the work items listed above.

1.3 Document outline

This document is structured in the following fashion: this section, *Introduction,* provides the background information that defines the purpose and intent of this report. Next, the *Methodology* section outlines the work-plan that was executed in the performance of this task. Next, the *Progress* section outlines the progress to date in the context of the over-arching objectives outlined in the scope section. Section 4, *Analysis of insurgent tactics*, details the current understanding of insurgent IED tactics. Section 5, *IED scenario generation software*, outlines both the current, proposed design for the scenario generation software, and the prototyping progress to date. Finally, the *Next steps* section summarizes the current status of this work and suggests priorities for future activities.

2 Methodology

In the execution of this task, three streams of work were performed in parallel.

First, a "Team" Cognitive Task Analysis (CTA) of insurgent activity was developed in which the overall goal of the insurgent team is to conduct an IED attack. This CTA was developed from government furnished information on patterns of insurgent IED activity. To that end, information on insurgent IED tactics was consolidated and reviewed to provide the analyst with an understanding of task organisation and task flow. The insurgents' roles in the tasks were categorized as: commander, builder, emplacer, spotter, trigger person, exploiter, transporter, and financier. The analysis of insurgent tasks by role focused on defining the associated human decision making steps, terrain requirements and equipment requirements needed to carry out simple and complex IED attacks. The CTA must be validated with Subject Matter Experts (SMEs) to ensure that it is an accurate representation of the tasks in terms of the activities performed, the order in which they are performed, and to provide additional data about the tasks. The CTA was based on the Canadian Forces' (CF's) body of knowledge on patterns and indicators observed in IED activity in recent deployments.

Second, the CF body of knowledge and the CTA were used to outline the requirements, architecture and preliminary design of the required scenario generation software. The knowledge of insurgent IED tactics was used to develop a picture of the technical requirements for the scenario generation software. Potential users of the software were identified and typified. The role of the scenario generation software in the larger scope of the IED awareness training project was outlined. This initial picture of the software requirements was used to develop an initial software architecture. Where possible, the requirements and architecture were used to outline the design of components of the software solution.

Third, in parallel, as central aspects of the required software functionality became apparent, prototyping activity was initiated to prove the concept of aspects of the scenario generation software, research potential tools and frameworks, and gain a better understanding of the technical complexity and information requirements demanded by the proposed design.

Given the timeline of this task, these three streams of work proceeded in parallel. The intent was for developments in one area to influence the focus of the other work. For instance, to allow the investigation of technical feasibility to influence the focus of the CTA development, and vice versa, to allow the priorities elucidated in the CTA to influence the focus of the prototyping activities.

3 Progress

This section provides an outline of the progress to date on this task. The outputs of this work, including the analysis of insurgent tactics, the architecture of the of scenario generation tool, and the results of the prototyping to date, are summarized in the following sections.

The initial activity of the task was to review the information available on insurgent tactics [1, 2, 3, 4, 5, 7, 8, 10, 11], and to produce an initial assessment of the requirements for software to be developed on this task. This initial analysis indicated that the initial focus on the mechanics of scenario definition within Bohemia Interactive's Virtual Battle Space 2 (VBS2) might limit the applicability of the results of this task. First, a number of the constraints observed were specific to this training tool, so working with these limits might constrain the applicability of work performed to other current or future CF training environments. Second, the work required to create the specific elements of insurgent tactical behaviour within VBS2 potentially overlaps with the mandate of the Land Software Engineering Centre (LSEC), and without further coordination with LSEC, development effort might be duplicated.

Based on this initial analysis, it was decided to focus the execution of this task on the development of the IED scenario generation process. That is, to focus on:

1. The capture of an analytical understanding of Insurgent tactics in the form of a Cognitive Task Analysis (CTA),

2. The definition of an architecture and design for the development of a useable scenario generation tool, and

3. The prototyping of a software capability to flexibly exploit the knowledge elucidated in the insurgent tactics CTA.

In this regard, Figure 1 below outlines the role of a scenario generation tool in the CF C-IED preparation process.

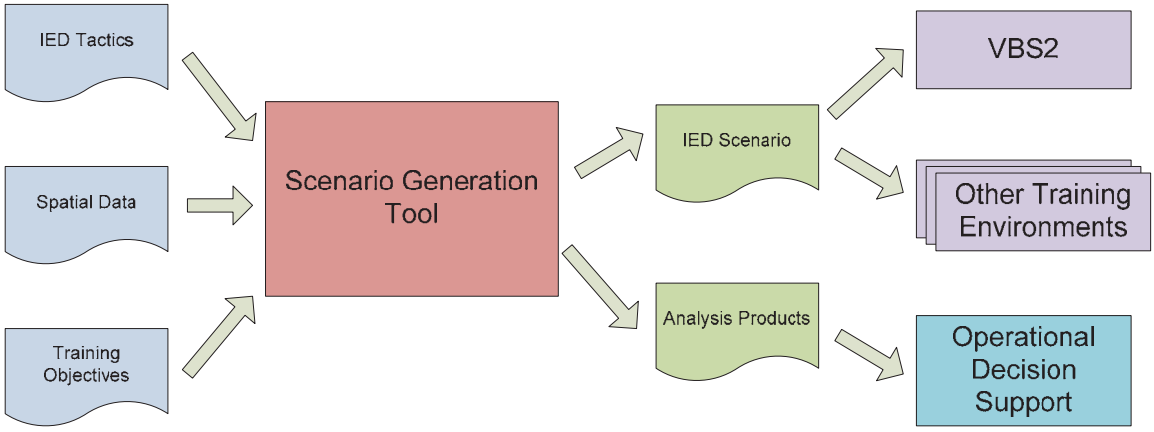

Figure 1: The role of a scenario generation tool in the C-IED process.

From the training perspective, the scenario generation tool operates on knowledge of insurgent tactics, spatial data describing an area of operation, and training objectives, to produce a definition of an

insurgent IED attack scenario. The scenario generation tool must be flexible and oriented to workflows that support the needs of the end-user, who may use the tool in different fashions at different times. The end-user and these workflows are discussed in detail in the *Requirements* section of this document.

The scenario generation tool must produce a representation of a tactically valid IED attack scenario that meets the specified training objectives. This scenario must include a definition of the elements involved in the scenario, their locations in the area of operation, as well as the details of how the execution of the attack will play out over time. The information in this scenario definition must be sufficient to specify that characteristics of a VBS2 mission and the use in other CF training environments, including live training must be considered. Further, it was noted that the capability required to analyze geospatial data to construct valid insurgent attack scenarios may have applications in analysis and the production of analysis products to support operational decision making.

Given the expanded role of the scenario generation tool defined above, configurability and flexibility in a number of areas were seen as key technology drivers for this activity:

- User workflow flexibility: based on the training objectives and tactical knowledge of the user (prototypically a CF trainer), the user inputs could vary from specifying the required attack outcome (i.e., damage, casualties, etc.), to specifying the capabilities of the insurgency, to specifying the location of the attack, to specifying the tactic to be employed, and the scenario generation tool would be required to assess plausibility around the constraints the user provides,
- IED tactics representation flexibility: the body of knowledge on insurgent tactics is complex, imprecise and evolving as insurgent tactics evolve, and
- Geospatial information flexibility – operational locations and potentially the availability of geospatial data will vary greatly, and the scenario generation tool should have the flexibility to perform to the greatest extent possible given variable spatial data inputs.

While it was initially understood that flexible geospatial analysis would be a significant component of the required software, the results of the initial analysis suggested an architecture with knowledge management, explicit representation of insurgent tactical knowledge, and flexible querying of this knowledge as a significant, additional component of the scenario generation tool. The scope of the prototyping activity was expanded to investigate techniques for managing tactical knowledge in a computationally addressable form.

The current status of the CTA of insurgent activity, the scenario generation tool requirements, analysis and design, and the prototyping activities are summarized in the following sections. Work is required to better define the target end users for this application, and to develop a user workflow, interaction and interface that matches their needs. The required work in this area is described in the *Next steps* section of this document.

4 Analysis of insurgent tactics

This section describes the task analysis [14], the human behavioural elements required for simulation, and the traceability of the analysis to the IED scenario generation software.

4.1 Task Analysis

The "Team" Cognitive Task Analysis (CTA) of insurgent activity in which the overall goal is to conduct an IED attack consists of eight tasks at the coarsest level of description. All top level tasks are not necessarily active in every scenario and the order of execution of tasks may vary from scenario to scenario. The top level tasks are:

4. Plan,
5. Develop and maintain shared situation awareness,
6. Ingress to IED site,
7. Emplace IED,
8. Actions on,
9. Egress from IED site,
10. Exploit IED attack, and
11. Abort mission.

An overview of the top levels of the analysis is provided next in the form of left-right diagrams. The task analysis file (Zobarich, 2011) provides the complete analysis to date which includes human behaviour requirements and traceability of the analysis to the IED generator software. Task Architect software is required to view and edit the file. A trial version is available for download at: www.taskarchitect.com.

It should be noted that the task analysis represents a somewhat idealized reconstruction of all the tasks and subtasks that would be involved in successful IED emplacement by insurgents, based on the CF body of knowledge that was made available to this project. That is, the CTA does not necessarily represent every task and sub-task that every IED emplacement cell would perform, or the exact order in which the tasks would be performed. However, the CTA captures our best reconstruction of the different tasks that could be involved across the range of IED emplacements observed by CF SMEs, as well as the logical relationships between them. We feel this provides a solid basis for generating a wide variety of specific instances of IED emplacements in the automated emplaced threat scenario generation tool.

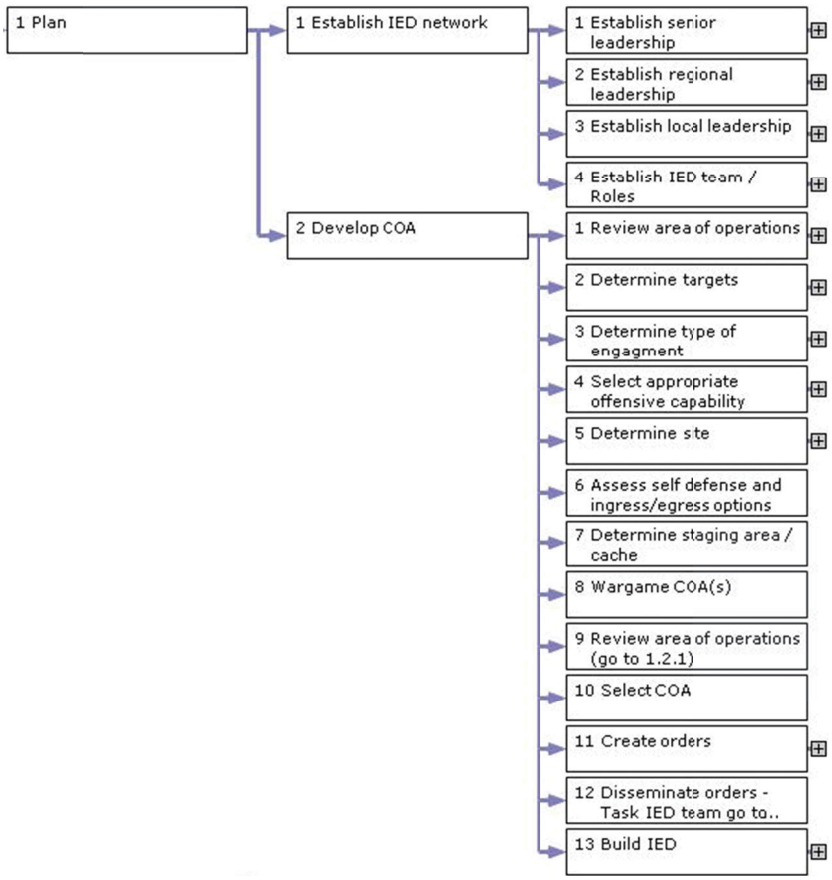

Figure 2: Overview of Plan

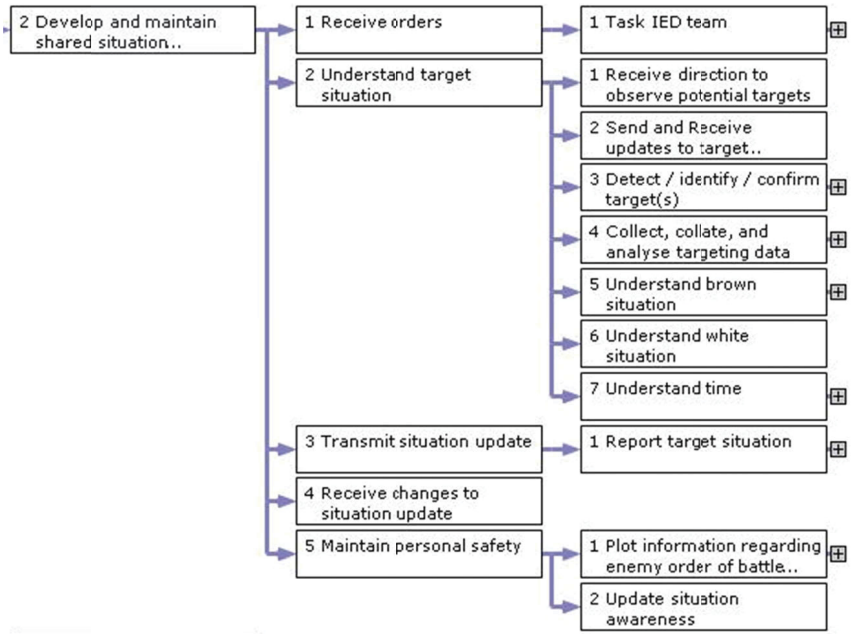

Figure 3: Overview of Develop and Maintain Shared Situation Awareness

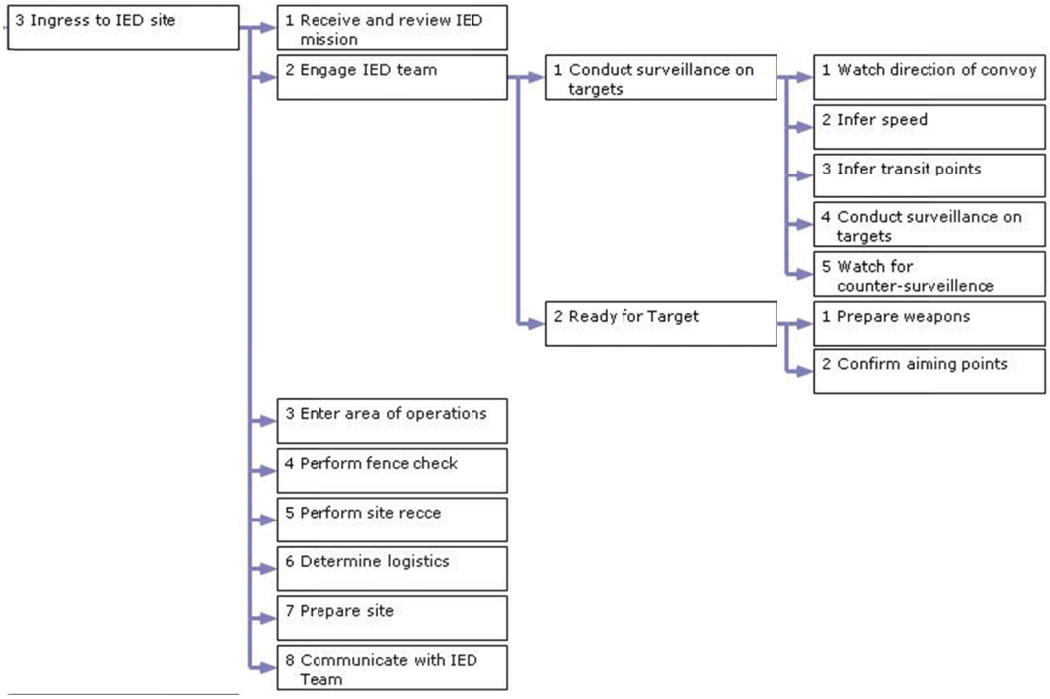

Figure 4: Overview of Ingress to IED Site

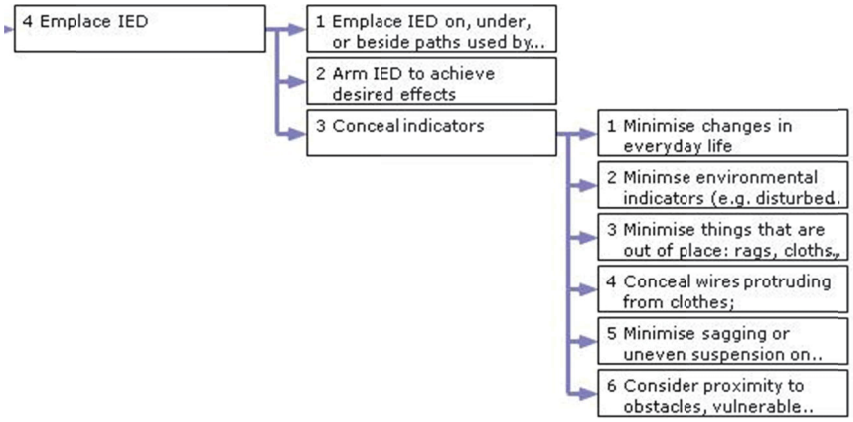

Figure 5: Overview of Emplace IED

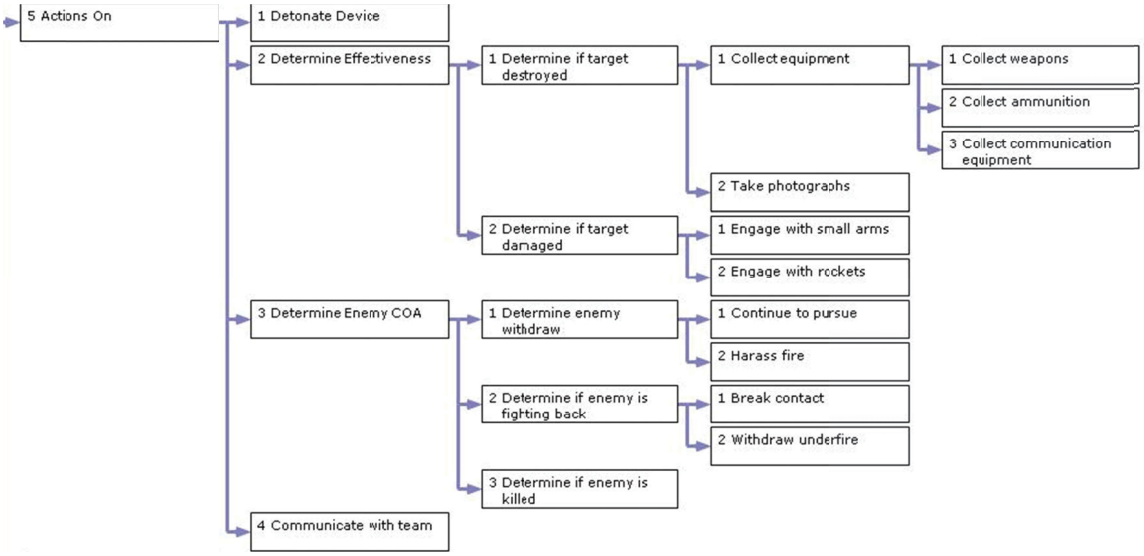

Figure 6: Overview of Actions On

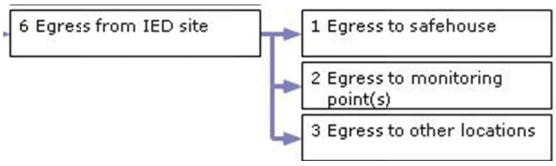

Figure 7: Overview of Egress

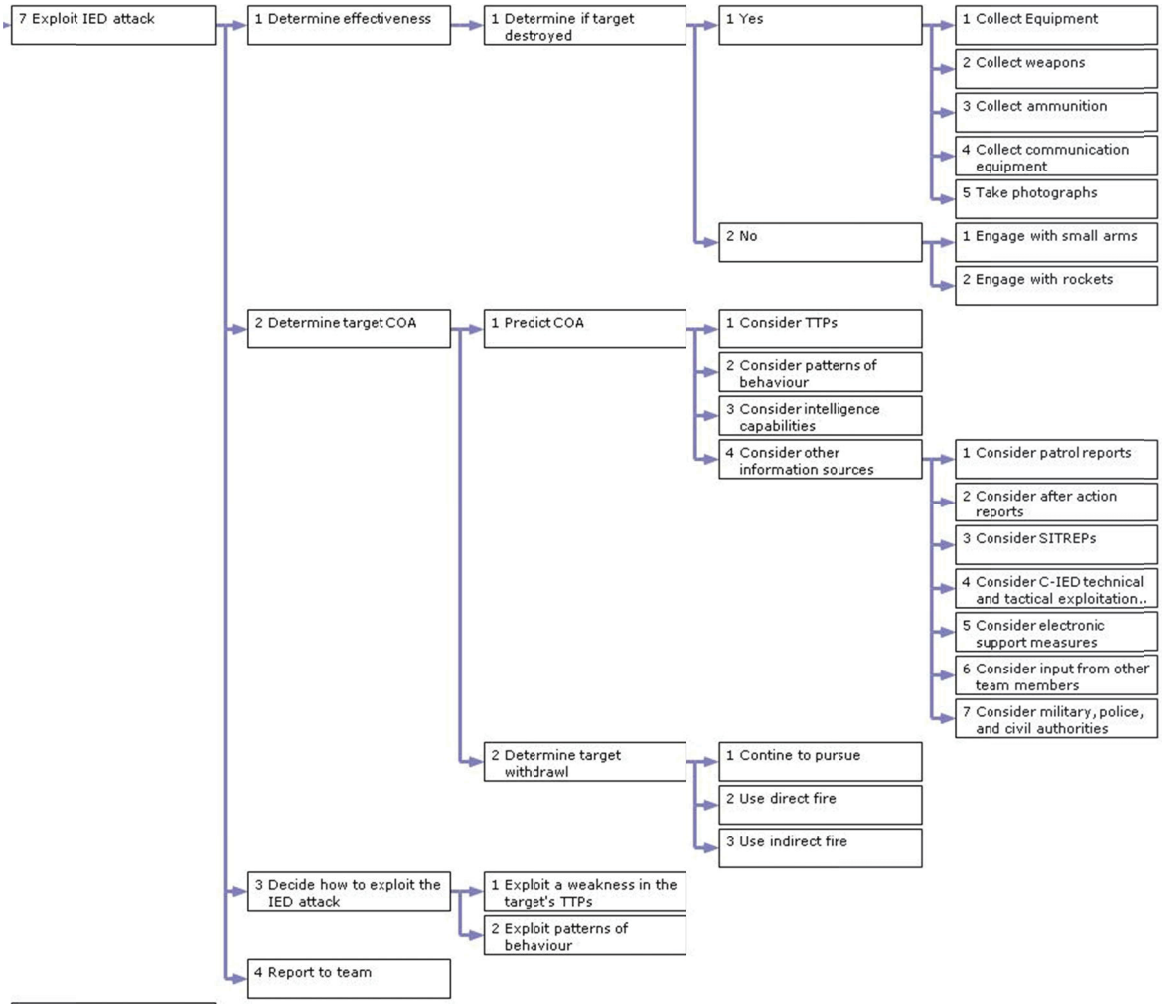

Figure 8: Overview of Exploit IED Attack

4.2 Human Behaviour Requirements

The lowest level, or leaf, tasks in the task analysis describe the behaviour or activity associated with the task. A tabular analysis lists the analysis of tactics, roles involved, composite terrain features, equipment and decision making. An example of the progress to date on the tabular analysis is pictured in the figure below.

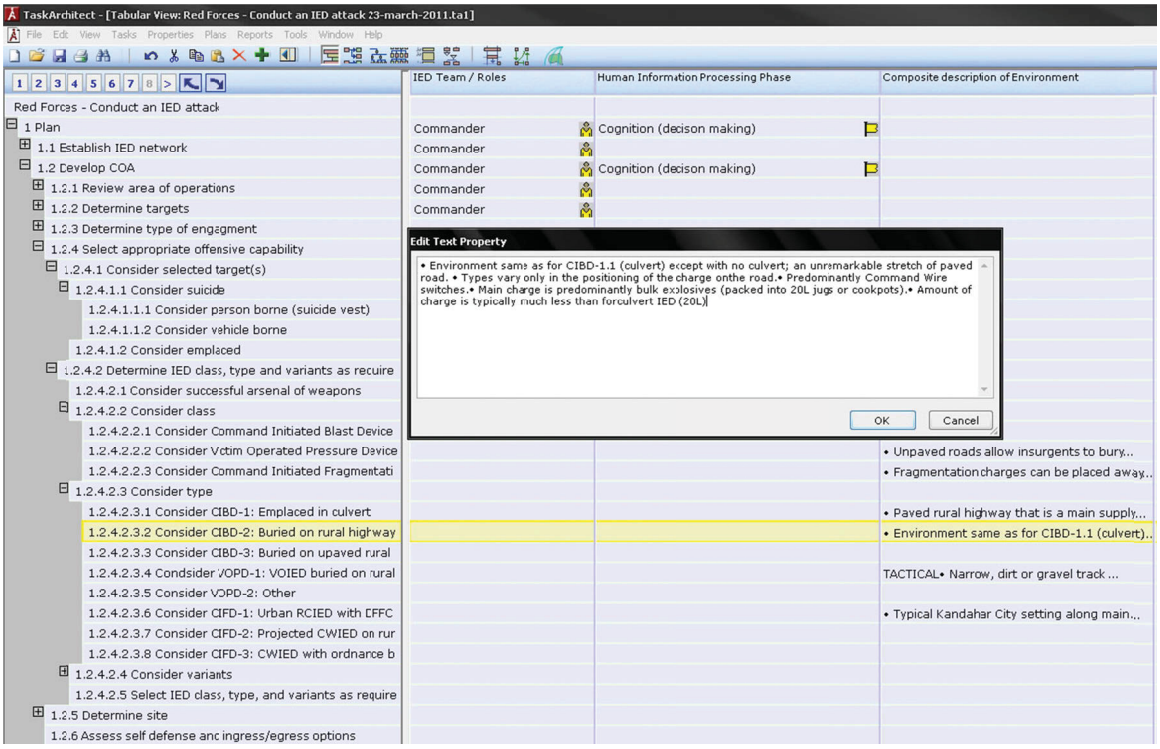

Figure 9: Example: Tabular Task Analysis of Insurgent Roles, Human Information Processing Phase, Terrain Features, and Equipment

4.3 Traceability

A framework has been established to capture the traceability between the task analysis elements and requirements for the IED scenario generation software. Additional columns in the task analysis file have been created and partially populated to allow attribution of tasks with ontology, user, and graphical user interface requirements traceability. Using comma separated value export as an intermediate format, formal requirements traceability could be established with requirements management tools if required.

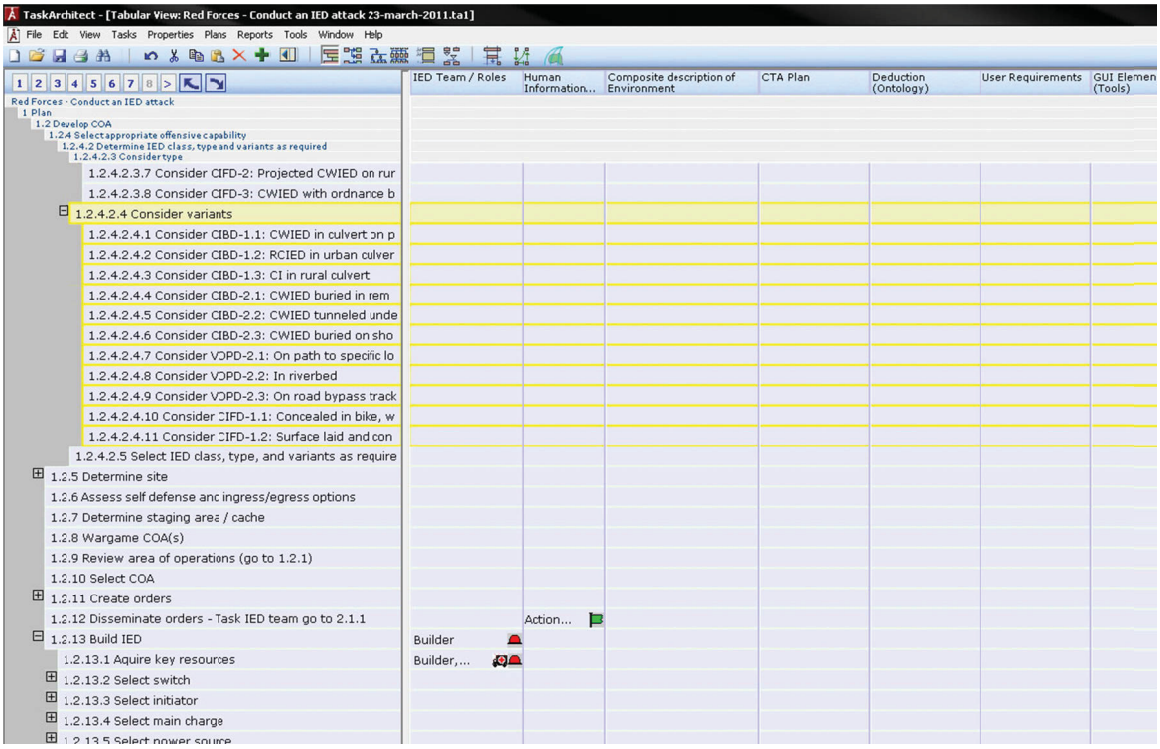

Figure 10: Mapping of the CTA plan, deduction (ontology), user requirements, and GUI elements (tools)

5 IED scenario generation software

This section defines software intended to support Canadian Forces (CF) users in the production of insurgent Improvised Explosive Device (IED) scenarios. The following subsections progress from the definition of the high level goals of this software, to the definition of the high level requirements for this software. This basis is used to outline an architecture for the system, which decomposed the software into components, and allocates requirements to those components. Analysis is performed to explain the implications of the knowledge captured in the insurgent Cognitive Task Analysis (CTA) on the requirements of the software components. An initial high level design is presented for these components. Finally, the progress of prototyping activities for these components is reported.

5.1 Goals

The vision of this work is to enable CF soldiers to make best use of their current systems to detect IEDs and assess the level and nature of IED threat in the operational environment (DRDC 2010).

The approach of this work is to improve the fidelity and effectiveness of IED awareness training through exposure of trainees to scenarios that embody tactically relevant IED threat scenarios and allow trainers to control scenario outcomes to provide a structured training environment.

With this basis, the goals of this software are as follows:

- to produce tactically relevant IED threat scenarios,
- to create effective training content for current simulation based training environments (such as VBS2),
- to plan for the creation of effective training content for tools and mediums of training or operational uses of analysis of insurgent attack tactics,
- to be easy to use by the end user community,
- to be applicable across any potential area of operations, including domestic operations,
- to make effective use of available GIS data,
- to enable the end user (the trainer) to get scenarios, understand scenarios and understand the quality or limitation of scenarios,
- to enable the end user to control the complexity and outcomes of the scenarios so as to be able to structure training with specific objectives and progressions, and
- to enable the transition of scenario definitions to executable training scenarios in selected simulation based training environments.

A limitation of the work reported in the following sections is that limited analysis has been performed typifying the end user – the CF C-IED awareness trainer. This limitation is detailed in the following sections, and steps to address this limitation are outlined in the *Next steps* section.

5.2 Requirements

This section outlines the high level requirements for the scenario generation software. First, the users of the software are defined. Next, use cases outline software stories of the normal ways in which the users will interact with the software. Finally, the software interfaces to the scenario generation software are described. The interfaces define the fashion in which the scenario generation software will depend on, or make use of data or other software and in what fashion will the scenario generation software create data or interface with other dependant software.

In the case of the IED attack scenario generation software, two forms of more detailed analysis fall out of the high level requirements definition. First, the requirements to accurately portray IED tactics captured in the CTA leads to analysis of the software implications of the information contained in the CTA. This analysis is outlined in Section 5.4, *Analysis*, below. Second, the definition of the users in Section 5.2.1, below, should lead to a more detailed analysis of the user interactions and interface required to support the objectives of these user groups. This analysis has been deferred until more information can be collected on the end-user community and their needs for this software. Steps to collect this information are outlined in the final *Next steps* section.

5.2.1 Users

Three main groups of user have been identified for the scenario generation tool: two current, and one future. The current users of the tool are 'CF IED awareness trainers' and 'defence scientists'. A potential future user is the 'operational analyst'.

5.2.1.1 Canadian Forces IED awareness trainer

The CF IED awareness trainer is responsible for defining and executing simulation-based training. They are responsible for preparing soldiers to manage IED threats encountered during deployment. They are responsible for exposing soldiers to simulations of plausible IED attacks of increasing complexity and instructing soldiers on the indicators of IED threat, and appropriate response to IED incidents.

The IED attack scenario generation tool must enable the CF IED awareness trainer to create IED attack scenarios that are tactically valid, meet the training objectives, and conform to the geography of real or virtual training areas.

5.2.1.2 Defence scientist

The Defence Scientist may make use of the tool for experimental or investigative purposes. The Defence Scientist may use this tool to generate training scenarios and conduct trials to measure training effectiveness.

5.2.1.3 Operational analyst

It has been recognized that in the future, an operational analyst may be able to make use of IED scenario generation software to better understand operational IED threat and risk. The operational analyst may be responsible for ensuring the scenarios produced by the IED attack scenario generation tool are tactically valid. The operational analyst may compare the scenarios generated by the tool to real world data. The

operational analyst may alter the performance of the tool to correct errors in validity or to update tactics based on the evolution of the insurgent threat. No analysis has been performed in relation to this potential user.

5.2.2 Use Cases

The outlines of the key use-case for the scenario generation software are as follows, labelled by the key user:

- Trainer - prepare scenario: the trainer produces a plausible IED attack scenario:
  - Select map data: the trainer selects the map data set to use for scenario creation,
  - Select area: the trainer selects the area of interest for the scenario:
    - Select area by rectangle,
    - Select area by polygon, and
    - Select area by query.
  - Define constraints: the trainer selects the constraints for the scenario, which could be where the attack occurs, what are the capabilities of the insurgents, or what the is the desired outcome of the attack,
  - Define blue force patterns: the trainer defines the blue force routes and forces for the scenario,
  - Validate inputs: if there are issues with the inputs (i.e., over-constrained or under-constrained) the tool must review the issues with the user in user language and guide the rectification of the issues,
  - Get scenario components and locations: the tool presents to the users what components (insurgent actors, devices, debris, etc.) are part of the new scenario, and where they are located, and
  - Understand how scenario will play out: the tool presents the logic of the scenario to the user to explain how the scenario will be played out.
- Trainer – prepare scenario based on training objectives: the user has a focus on accomplishing certain, specific training objectives,
- Trainer – prepare scenario based on insurgent capabilities: the user has a focus on exposing the trainees to specific tactics or a specific threat,
- Trainer – prepare scenario based on location: the user has a focus on the attack occurring at a specific location,
- Trainer - evaluate scenarios: the user produces or imports or loads a scenario and wants analysis to support the evaluation of the scenario:
  - Select scenario components,
  - Select component locations,
  - Define scenario events: the user defines the scenario,

- Understand quality of scenario: the tool presents a quantitative assessment of the scenario, and
- Understand reasons for high or low quality: the tool provides specific reasoning for the quantitative assessment.

- Scientist – validate knowledge base: the scientist interacts with the execution of the scenario generation tool to assess the validity of the steps performed and the overall scenario:
  - Specify condition,
  - Observe tool intermediate output,
  - Observe steps in deductions and spatial processing, and
  - Manage test cases.
- Scientist – update tactics: the scientist interacts with the knowledge base and the processing performed by the tool to alter or extend the tactics information and to change the nature of the scenarios that are generated:
  - alter knowledge base, and
  - alter spatial processing tools.

5.2.3 External Interfaces

Currently, the scenario generation tool has been formulated as a standalone tool that operates on file based source data to produce file based output. No software interfaces to other applications are anticipated.

As input, the tool will need to make use of GIS data. GIS data from different areas may conform to different content standards, and certain software operations may not be possible if certain items are not present in the source GIS data. It is anticipated that GIS data will be available in ESRI ShapeFile format. The baseline for the data content standard will be the content of Vector Map (VMAP) level 2, similar to a 1:50,000 topographical map. The baseline for feature and attribute classification is the Digital Geographic Information Exchange Standard (DIGEST) Feature and Attribution Coding Catalogue (FACC) [9]. The Mapping and Charting Establishment data for Wainwright conforms to this standard. The source will contain point, lineal and areal representations of features in vector data layers, and raster representations of elevation in a digital elevation model. The quality of the scenarios generated will be directly affected by the quality of the GIS source data.

The tool will produce scenarios as output. A scenario is composed of the definition of a variable number of components, the locations of these components and the logic or rules by which these components will respond to blue force activity to execute the IED attack. The tool should represent the scenario in an open and accessible format to enable the re-use of scenario in the tool itself, and generic capabilities such as mapping software. In addition, specific exporters will have to produce formats for specific training devices, such as mission files for VBS2. Note that the mission files may require specific, custom functionality to be installed in VBS2 to enable the execution of the IED scenarios.

Military Scenario Definition Language [12] should be considered as an export format. MSDL import is gaining support in military simulations such as the United States Department of Defense OneSAF

product. Export of the scenarios in a human-readable, report-like format should also be considered to enable use of the information from an analysis perspective.

5.3 Architecture

This section outlines the system architecture for the IED attack scenario generation tool. Figure 11, below, depicts this architecture.

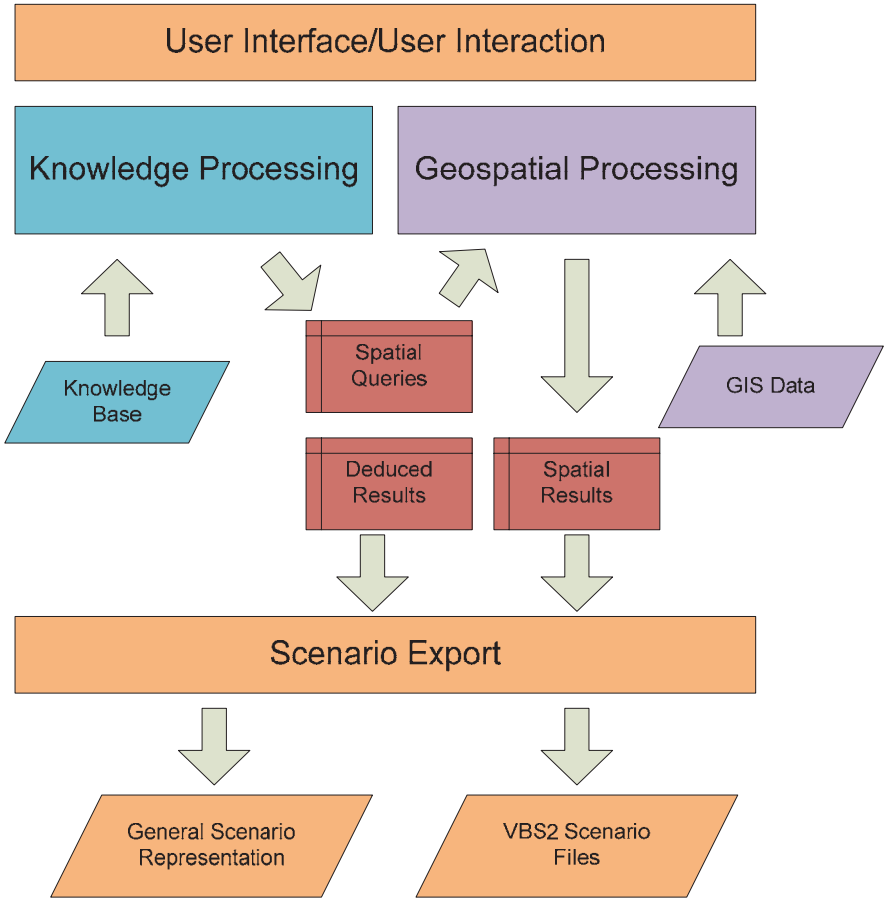

Figure 11: Notional system architecture for the scenario generation tool.

The execution of the software processing is controlled by a User Interface (UI) layer. The user interface is responsible for presenting information to the user, accepting input from the user, and providing a structured interaction or workflow to the user. The user interface may be specialized to the different user categories.

The UI controls the execution of a knowledge processing engine, a geospatial processing engine and the scenario export component. The knowledge processing engine is responsible for working with a knowledge base representing insurgent tactics and producing both deduced results (the scenario will be a victim operated attack on a patrol route) that are part of the required scenario, and intermediate spatial queries or requests (where are the gravel roads on the patrol route?). The geospatial processing engine operates on Geographical Information Systems (GIS) or 'map' data to produce the spatial results of the

spatial queries. Together, these two pieces of information, the deduced results and the spatial results are the basis of the scenario definition.

The scenario export component is responsible for saving this scenario definition information in standardized forms, and as required input information for training simulation environments such as VBS2.

5.3.1 User Interface

In the initial assessment, the CF trainer user cannot be expected to be familiar with GIS systems, insurgent tactics or with the geographic specifics of an area of operations. Their primary objective is the production of scenarios driven by training objectives. They need to generate, modify and use plausible IED attack scenarios.

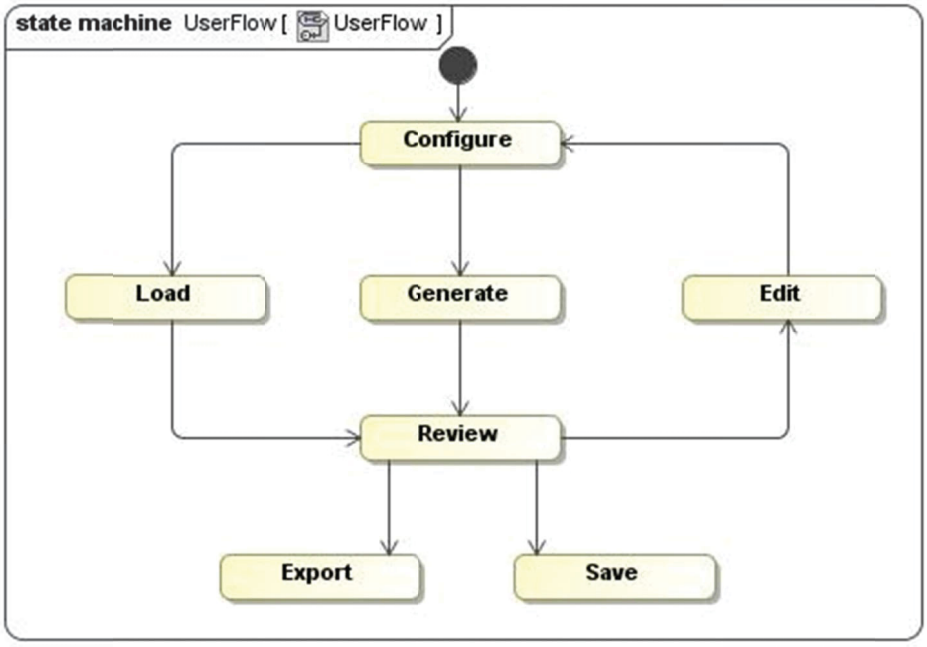

Figure 12: State chart of user interaction with the scenario generation tool.

With this is mind, a linear generation process (wizard) pattern has been selected as the initial model for the trainer's interaction with the scenario generation software. The primary interaction path is as follows:

1. The user launches the application,

2. The user configures the boundaries of the scenario they want to generate, for instance, selecting an area of interest and selecting the tactic,

3. The user triggers the software to generate a scenario,

4. The user reviews the scenario, and

5. The user exports the scenario for use in the training environment.

Optionally, the user can:

- On review of the scenario, edit the scenario, returning to the configure process, to regenerate a modified scenario,
- Save the scenario in an archival format, or
- Load a previously saved scenario and be returned to the review process, to export or edit the scenario.

This user workflow will have to be validated against the expectations of the user group in further work. A different user interface may be used for the scientist user, who might benefit from a GIS-style map and tool interface that allows visibility into the working of the generate process. The architecture isolates changes between the scientist and trainer workflows to the UI component – the same processing engines will be used to support both interfaces.

5.3.2 Knowledge processing engine

The knowledge processing engine is embedded in the Generate stage of the trainer user's workflow. The purpose of the knowledge processing engine is to map the user's higher level, tactical/military interaction with the system into lower level, concrete, actionable assertions and constraints that can be used to generate the specifics of the scenario. This component must produce the list of components of a plausible scenario and derive the relationships that must be preserved. For instance, if the user specifies that they want a wire command initiated attack along a certain route, it is the purpose of the knowledge processing engine to determine that this means a device must be concealed or buried along the route and the trigger person must be concealed within a certain range of the device, with visibility of the route, etc.

A subset of this information is the input to the geospatial processing that must be performed (see the next section).

The knowledge processing engine operates on a standardized representation of insurgent IED tactics – the knowledge base. The scientist user must be able to edit the knowledge base to support updates as insurgent tactics evolve and to support correction during validation. Also, the engine most likely must be able to produce intermediate results to support 'debugging' to enable scientific visibility into processing to support validation.

The knowledge base will be represented in a standard form, such as the Web Ontology Language (OWL), Prolog or the Resource Description Framework (RDF). The basis of the knowledge processing engine will be a decisional logic reasoner or subset such as SWI-Prolog or Racer Pro.

5.3.3 Geospatial processing engine

The geospatial processing engine is also embedded in the Generate stage of the user's workflow. The geospatial processing engine provides a toolbox of kernel, composible, geospatial processing tools that are configured and used in accordance with the output of the knowledge processing engine to produce the geospatial characteristics (location of device, location of other components, etc.) of the required scenario.

The engine loads available GIS data under the direction of the user and uses information from the knowledge processing engine to map the specific data structure and attribution to the classes and properties defined by the knowledge base and used by the geospatial processing tools.

The core functionality of the geospatial engine is to calculate the results of the required spatial queries, and then to combine the results to provide final output of the scenario component locations. For the scientist user, the tools will have to provide intermediate results to support validation.

The geospatial tools will make use of a standard GIS development environment to enable the packaging of the scenario generation tool as a custom mapping application or as an extension to an existing mapping suite. The geospatial processing engine prototype has been built using ESRI ArcObjects.

5.3.4 Scenario export component

The scenario export component saves a standardized representation of the scenario components (type, details, and locations) and the time/tactical details of the scenario. This component will export to a standard, open archival format, as well as to specific platforms, such as VBS2 mission files. In this case, this will require contribution from VBS2 support or development to implement the runtime aspects in VBS2 to play out required scenario elements that are not natively supported by VBS2.

This component will support the extension to export to new platforms as required, and the nature of the archival format should be based on GIS standards to allow use of the scenario material in other standard tools.

5.4 Analysis

The analysis reported in this section is the examination of the CTA and description of insurgent tactics to produce a list of the technical requirements for the knowledge processing engine and the geospatial processing engine. Significant analysis must also be performed to translate the user needs into technical requirements. This activity is pending further interaction with the targeted end user group.

5.4.1 Requirements for ontology engine

First, a taxonomy for the IED tactics CTA was produced. The taxonomy defines the authoritative terms for all the objects and items discussed in the CTA. Any items discussed in the analysis of the tactics should be classified by the taxonomy. An initial taxonomy for the IED attack CTA is shown below in Figure 13.

Figure 13 The initial IED taxonomy

The taxonomy is the 'nouns' of the knowledge base, and is a class hierarchy with the possibility of multiple inheritance.

Next, an initial list of the object properties – the relationships between taxonomy items (classes) was formed. Object properties are the 'verbs' of the knowledge base and define the ways classes can be related. These relationships form the basis of the classification of items by inference. The figure below highlights an example object property from the OWL based knowledge base prototype, viewed in Protégé.

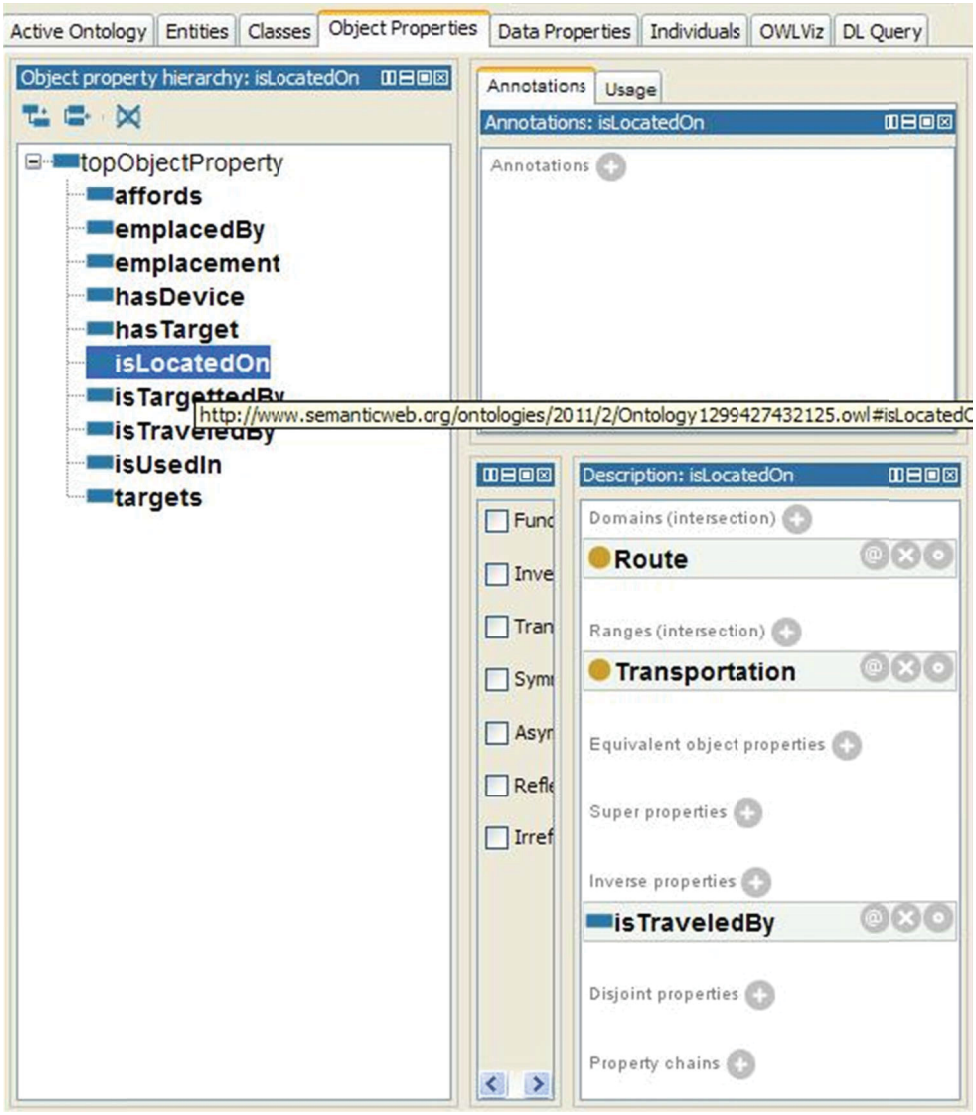

Figure 14: An example object property from the prototype knowledge base.

Object properties can be explicitly restricted in their domain and range of applicability, and can be attributed with the logical properties that further control the deductions that can be made. In the context of noun-verb-noun triples, the first noun is the domain and the second noun is the range. Figure 14 above, as example, can be read: a 'Route' (such as a main supply route) isLocatedOn a 'Transportation'

(such as a gravel road). In the list below, example relationships are listed in pseudo-code and briefly described. These relationships are derived from the documents listed in the *References* section:

- Target travels Route: all potential targets travel along linear paths called routes,
- Attack isLocatedOn Route: the attack must be located on a route,
- AttackLocation isTraveledBy Target: the location of the attack must be traveled by the intended target,
- VictimOperatedDevice emplacedBy Burial: victim operated devices are emplaced by being buried at the attack location,
- SoftSurface affords Burial: anything that is classified as soft surface can have devices buried in it,
- Culvert affords DeviceConcealment: devices can be hidden in culverts,
- Vehicle affords Concealment: devices can be hidden in vehicles, and people can hide in vehicles,
- Bicycle affords DeviceConcealment: devices can be hidden on bicycles,
- Debris affords DeviceConcealment: devices can be hidden in debris,
- ComandIntiatedAttack hasDevice CommandInitiatedDevice: if a command initiated device is used, it is a command initiated attack,
- VictimOperatedAttack hasDevice VictimOperatedDevice: if a victim operated device is used, it is a victim operated attack,
- Blast targets Target: blast charges can be used to attack all types of targets,
- Frag targets Person: fragmentation charges are only used for antipersonnel,
- CanalizingFeature favours AttackLocation: attacks are more likely where canalizing features exist,
- RoadBend affords SpeedRestriction: bends in the road restrict the speed of targets,
- SteepHill affords SpeedRestriction: steep hills restrict the speed of targets,
- RoughTerrain affords SpeedRestriction: rough terrain restricts the speed of targets,
- SpeedRestriction favours AttackLocation: attacks are more likely where target speed is restricted,
- ECM defeats RadioControlledTrigger: electronic counter measures can defeat radio control triggered devices,
- AttackType hasPhase Emplacement: all attack types have an emplacement phase,
- CommandInitiatedAttack hasPhase Overwatch: command initiated attacks have an overwatch phase, and
- RaisedRoute favours CommandInitiatedAttackLocation: command initiated attacks are more likely where the target route is elevated.

A number of observations outlined in the IED material form more complex axioms, and may need to be further decomposed to avoid computational complexity in the reasoner:

- if non-Target travels Route, VictimOperatedAttack is not possible: victim operated attacks cannot be used on routes with traffic that is not to be targeted,
- Urban favours Frag: fragmentation charges are more likely in urban areas,
- RadioControlledTrigger is required by Frag: fragmentation devices are triggered by radio control,
- Remote favours VictimOperatedAttack: in remote areas, victim operated attacks are more likely,
- TriggerLocation isConnected to DeviceLocation by Trigger: the trigger connects the overwatch or trigger location to the device location,
- TriggerLocation isConnectedTo SafeHouse by EscapeRoute: the trigger person escapes from the overwatch location using an escape route,
- DeviceLocation isConnectedTo SafeHouse by EmplacementRoute: the device is carried to the attack location along an emplacement route,
- ConcreteCulvert requires ChargeSize > 50l: charges larger than 50l are required in concrete culverts,
- Rural favours Emplacement: device emplacement is easier in rural areas,
- Remote favours Emplacement: device emplacement is easier in remote areas,
- CommandInitiatedAttacks in Rural normally use WireTrigger: in rural areas, wired command initiated attacks are more common,
- CommandInitiatedAttacks in Urban normally use RadioControlledTrigger: in urban areas, radio command initiated attacks are more likely,
- WireTrigger is normally Buried: wire triggers are normally buried,
- Culvert:Depth less is better for AttackLocation: attacks are more likely in shallow culverts,
- Culvert:Grated is worse for AttackLocation: attacks are less likely where grates are installed on culverts,
- historyOf VulnerablePointSearches is worse for AttackLocation: attacks are less likely in areas that are searched more frequently,
- historyOf Attacks is better for AttackLocation: attacks are more likely in areas where attacks have occurred before,
- history of Target is better for AttackLocation: attacks are more likely where target travel patterns are established,
- for Blast, LocationUnder is better: blast charges are normally located under the target route,
- for Frag, LocationBeside is better: fragmentation charges are normally located beside the target route,

- RadioControlTrigger:Length is less than WireTrigger:Length: radio controlled triggers have less range than wire triggers,
- CoallitionTargets are more likely to have ECM, and
- normal ChargeSize is 20l.

Finally, equivalencies are defined. Equivalencies are classifications that can be made solely from object properties. That is to say, the object properties are sufficient to classify the instance. For instance, the observation "if it purrs, it must be a cat" states that having a purr is sufficient to classify something as a cat.

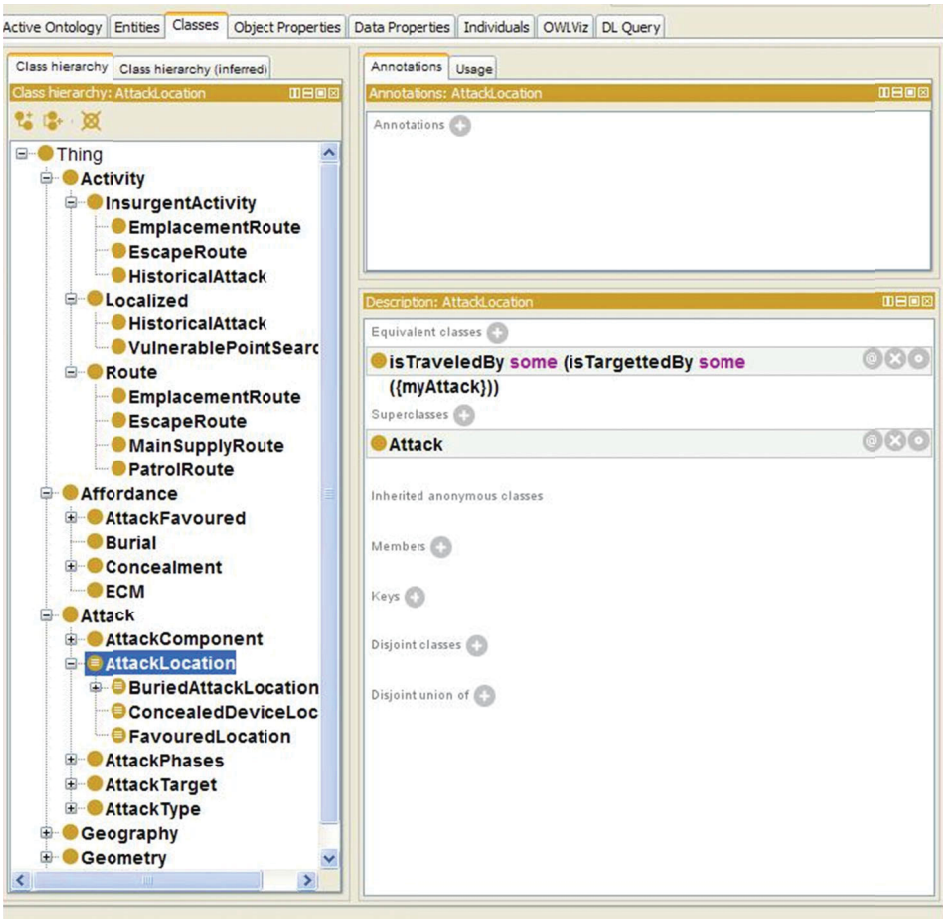

Figure 15: An example of equivalency in classification.

The figure above gives an example of equivalent classification. This example asserts that a potential 'Attack Location' is anything traveled by something that is targeted by 'my attack.' That is to say, the attack can only be located where the target travels.

The information above is the basis for an initial IED tactics knowledge base. The formation of this initial knowledge base and the use of a reasoner to operate on this knowledge base is described in the *Design and Prototyping* section 5.5.1, *Knowledge processing engine*.

5.4.2 Requirements for geospatial processing engine

The spatial relationships discovered in the analysis of insurgent tactics and the devices used in insurgent tactics were used to derive a list of the kernel geospatial operations required by the geospatial processing engine. These operations are listed below:

1. Find By Type

    a. Find By Feature layer/class: This tool finds all the layers by "Name" as input. The Ontology Engine will provide the layer name to search for. Ex. Find all the "roads" – The ontology would provide the name "roads" and the layer will be found by the tool.

    b. Find By Attribute: This tool finds all features in a layer given an attribute. The Layer and Attribute type will be provided from input from the ontology engine. Ex. Find all the gravel roads – The ontology engine would provide all the layers that contain "roads" and the attribute on "roads" that would indicate that they are "gravel".

2. Find By Location: This tool finds all features in an Area of Interest that is provided by the user. The user will provide an area as input, also the features and attributes on those features if required. Ex. Find all roads in rural area (Area of Interest) – The ontology engine would provide the "Area" to search on the map, "roads" as the feature layer/class and "rural" as an attribute on the map to search by.

3. Find By Proximity

    a. Simple Buffering: This tool finds all features from a point provided by the user that are buffered "x" distance away from that point. The input for this tool would be the "Point" indicated by the user, the feature layer/class that will be buffered and the buffer distance. Ex. Find all buildings within 200m of the device location – The ontology engine would provide the device location as the "point" on the map, "buildings" as the feature layer/class and "200m" is the buffer distance.

    b. Complex Buffering by Range: This tool finds all features within a range from a point provided by the user. Given a point, a minimum range x and a maximum range y, the tool would return features that where range is larger than x and less than y from the supplied point. The input for this tool would be the "Point" indicated by the user, the feature layer/class that will be buffered and the buffer distances. Ex. Find all buildings within 30m - 200m of the device location – The ontology engine would provide the device location as the "point" on the map, "buildings" as the feature layer/class and 30m to 200m as the buffer range distance.

    c. Complex Buffering: This tool finds features that are narrowed by other features on the map. The input for this tool would be the feature layer/class of the features you want to find, the feature class/layer of the features that could narrow the features you want to find, and how far they should be narrowed to. Ex. Find all roads narrowed to 3m by canalizing features such as buildings and bridges – The ontology engine would provide "roads" as the feature layer/class, "3m" and the narrowed width and "buildings" and "bridges" as the feature layer/classes that would be checked to narrow "roads".

d. Close Features: This tool finds linear intersections of features. The input for this tool would be the two feature layer/classes that will be analyzed for intersections. Ex. Find where streambeds cross roads – The ontology engine would pass "streambeds" and "roads" as the input feature layer/classes for processing.

4. Elevation

   a. Line of Sight: This tool finds if there is a line of site between two points on the map. The line of sight will be indicated by "red" or "green" on the map, "green" being full line of sight in that area and "red" being no line of sight in that area. The input for this tool would be the 2 points on the map. Ex. Is there line of site between the device location and building x – The "points" on the map indicating the locations of the building and the device will be given, the output will be shown on the map and the user will be able to see whether there is line of sight.

   b. ViewShed: This tool will provide areas where you have line of sight to a point on the map. The input for this tool would be the point on the map to get the information on. Ex. What are the areas that can see the device location – The device location "point" on the map will be used to calculate what areas are able to see this location. These locations will be highlighted to the user.

   c. Slope: This tool finds out where there are slopes on the roads (due to hills, etc...) in an Area of Interest. This input for this tool would be the Area of Interest, the feature layer/class that would be checking this data, and the elevation data for the Area of Interest. Ex. Find all road areas where the traffic will slow due to hills – The ontology engine would provide the "roads" feature layer/class and the Area of Interest. From this data the elevation data would be checked in line with the "roads" layer to find out where there are areas of slope that could slow down traffic.

5. Geometric Characteristics – Curvature: This tool finds features that are affected by "Curvature" in an Area of Interest. The input for this tool would be the feature layer/class and the Area of Interest. Ex. Find all road areas where traffic will slow due to curvature in the road – The ontology engine would provide the "road" feature layer/class and the Area of Interest. This tool will then look at the geometric characteristics of the roads in that area to look for curvatures in the road. The tool will highlight to the user the areas on the "roads" that would slow traffic due to curvature.

6. Geometric Intersection: This tool finds all features that intersect each other. The input for this tool is x feature layer/class(s). Ex. Find where there are soft surface roads with buildings that are within 200m from them – This example would make use of 2 previous tools (Find by Attribute to find all the soft surface roads (A new layer would be created from this output). The Simple buffering tool would be used to find all "buildings" that are within 200m of each of these "roads" (A new layer would be created from this output). The input for this tool would be the 2 layers created by the other tools in the process to highlight the intersections of these features to the user.

7. Routes: This tool handles route assessments. This tool needs further investigation. Examples:

   a. Are there suitable routes from the overwatch location?

   b. Are there suitable routes for retreat from attack location?

c. Are there any concealed routes around the attack location?

8. Statistical: This tool will handle statistical information about the Areas of Interest. This tool needs further investigation. Examples:

   a. Where have attacks occurred in the past?

   b. Who travels this route and how frequently?

   c. What are the known historical patterns of insurgent activity?

   d. Where are the safe houses in the area?

   e. Intel on locations

   f. Urban/Rural/Remote assessment from population density information

9. Topology: This tool will analyze the topology of an Area of interest. This tool needs further investigation. Examples:

   a. Connecting waypoints with a network

   b. Find road segments that are involved in a MSR specified by waypoints.

Some examples of spatial questions that will be answered by these tools are listed below:

- find Route: where does the target travel?
- find Transportation traveledBy Route: what are the particular transportation segments in the GIS data that the target travels on?
- find SoftSurface in Transportation traveledBy Route: which transportation segments traveled by the target have a composition that is in the set of things that are soft surface?
- find Culvert traveledby Route: what culverts are along the target's route?
- find Debris closeTo Route: where is there debris close to the target's route?
- find RoadBend in Transportation traveledBy Route: where are there road bends along the target's route?
- find CanalizingFeatures onBothSidesOf Route: where is the width of the target's route restricted by canalizing features?
- find DeviceConcealment closeTo Route: where are their items along the route in which devices can be concealed?
- find PersonConcealment minMaxDistFrom Route: where along the route are their features in which people can hide that are within a specified range of distances from the route?
- find PersonConcealment withLineOfSightTo Route: does the concealment location have a line of site to the target route?

- find PersonConcealment with EscapeRoute: is there an escape route from the potential concealment location?
- find DeviceLocation with EmplacementRoute: is there an emplacement route to the potential device location?
- find DeviceLocations closeTo Features: what features are around the potential device location?
- find SteepSection traveledBy Route: where are there steep hills along the target's route?

The table below gives a mapping between some of these tool use-stories and the individual tools.

Table 1 : Initial mapping from geospatial processing stories to geospatial tools

Use Example	**Find by class**	**Find by attribute**	**Find in polygon**	**Find by buffering**	**Find by intersection**	**Find by curvature**	**Find by slope**	**Find by LOS**	**Find by view shed**	**Find by route assessment**	**Combine processing**	**Heat mapping**
Find all roads	X	X										
Find roads travelled by the target	X	X			X						X	
Find all gravel roads	X	X										
Find … in rural areas			X									
Road segments from route					X							
What culverts are along route				X	X							
What stream beds cross the route					X							
Where are there clutter features				X								
Where are there canalizing features				X								
Possible overwatch features	X	X		X				X			X	
Where is target speed restricted				X		X	X				X	
Is there an escape route										X		
Is there an emplacement route										X		
History of attacks												X
History of patrols												X
History of convoy operations												X

Use Example	Find by class	Find by attribute	Find in polygon	Find by buffering	Find by intersection	Find by curvature	Find by slope	Find by LOS	Find by view shed	Find by route assessment	Combine processing	Heat mapping
How concealed is the escape route									X			
How well can overwatch see the route									X			

5.5 Design and prototyping

This section reports on the progress to date in the design and prototyping of the knowledge processing and geospatial processing engines.

5.5.1 Knowledge processing engine

The knowledge processing prototype [13] is an OWL ontology, constructed in Protégé and then with inferences made with the Fact+ reasoner. A taxonomy was defined that is the standard terminology for all classes of objects that can exist. Hierarchical relationships show how properties are shared. Multiple inheritance is possible. The figure below is a fragment of the taxonomy that shows that a gravel road is a type of soft surface composition, in addition to being a road.

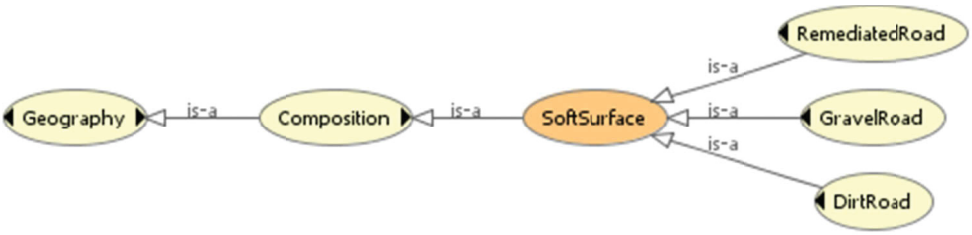

Figure 16: Example taxonomy of surface composition.

Next. Equivalencies were defined. These are defined rules that are sufficient to classify an object. ("All green things are plants..."). The example below shows the definition of SoftSurface as a class of things that affords (allows) burial.

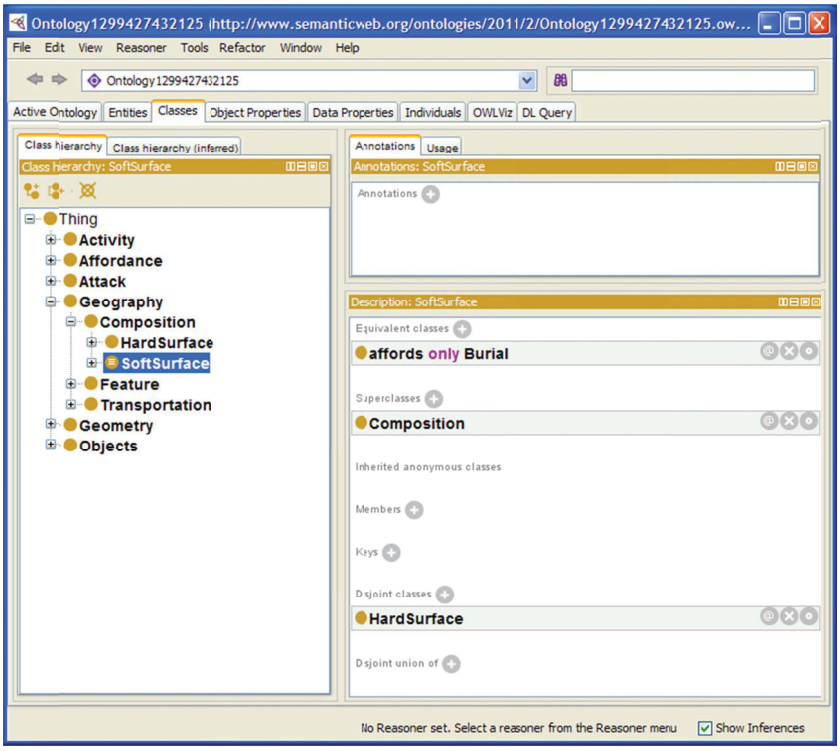

Figure 17: Example definition of class equivalency.

This example states that anything that you can bury things in, is a soft surface item. Conversely, if it is a soft surface item, you can bury things in it. At this stage, the explicit and implied classifications already lead to deductions in the classification of things. The figure below shows a subset of the inferred class hierarchy.

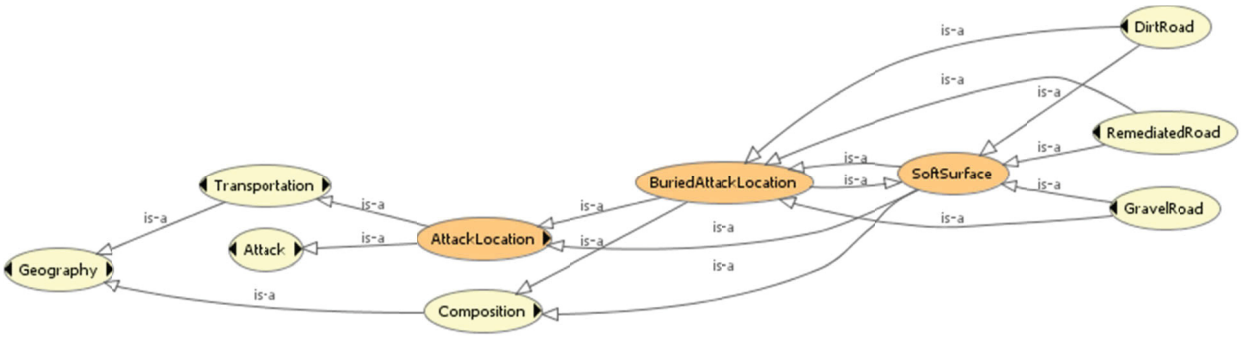

Figure 18: Example of inferred classification.

The classification relationships lead to the inference that therefore, a soft surface item is a potential buried attack location is a potential attack location.

Next, instances are defined. Instances are specific items – 'gravel road' is a class of things, while ''forest lane' is a specific item, that might be an example of a gravel road. In the prototype, instances are declared and relationships to other instances are defined.

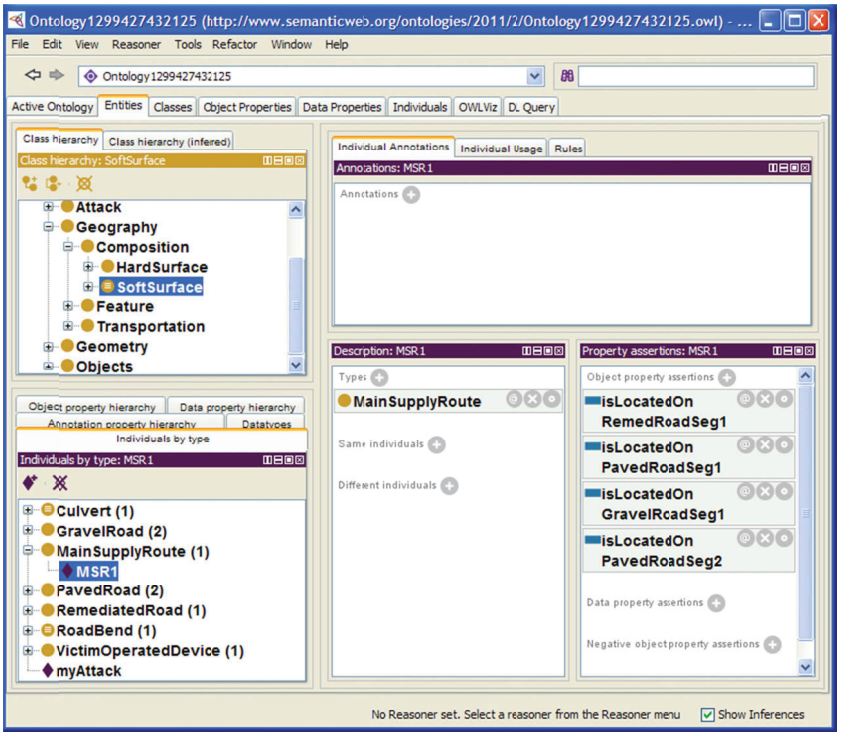

Figure 19: Example of a main supply route instance.

In the example above, the item called ‘MSR1’ is defined as a main supply route, and the route is defined to travels the road segments named ‘RemedRoadSeg1,’ ‘PavedRoadSeg1,’ ‘GravelRoadSeg1,’ and ‘PavedRoadSeg2.’

With these definitions, the inference engine can deduce further classifications of the defined items.

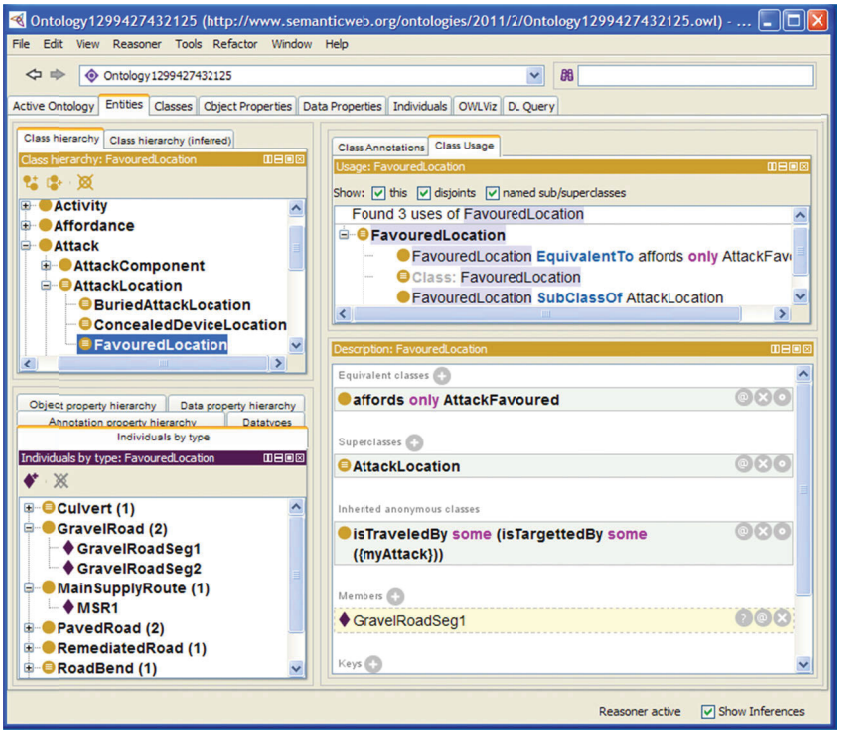

Figure 20: Example of deduced classification of defined instances.

Based on the assertions that include the facts that:

- GravelRoadSeg1 is a gravel road
- GravelRoadSeg1 is traveled by the target
- The attack device is a landmine
- The landmine is a victim operated device
- GravelRoadSeg1 is a tight bend in the road

The reasoner can classify GravelRoadSeg1 as a plausible and favoured location for an attack. In this fashion, the knowledge engine deduces the characteristics of attack component locations. The figure below details the inferred hierarchy of the prototype ontology.

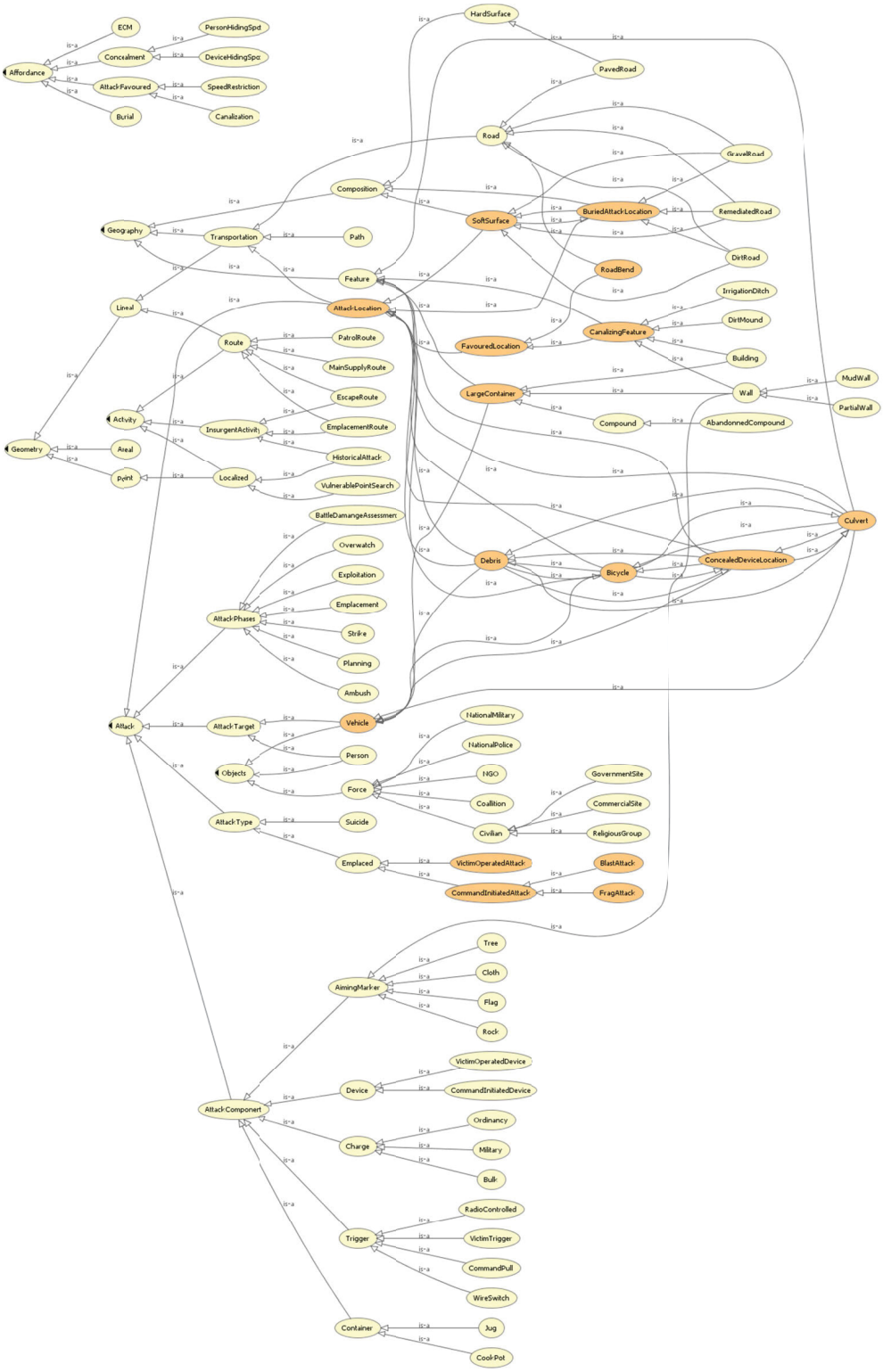

Figure 21: The inferred classification of the initial IED taxonomy.

Note, as the only knowledge that is represented about culverts, bicycles and debris is that they can all conceal devices, they are classified as equivalent classes (a bicycle is a culvert is a debris...). Within the scope of the ontology, no fact or rule has made a distinction between bicycles, culverts and debris, so they are classified as equivalent within the scope of the ontology.

5.5.2 Geospatial Engine

5.5.2.1 Background

Geographic Information System (GIS) technology is an integrated collection of computer software and data used to view and manage information about geographic places, analyze spatial relationships, and model spatial processes. GIS provides a framework for gathering and organizing spatial data and related information so that it can be displayed and analyzed. GIS gives you tools to analyze your data and see the results in interactive maps.

5.5.2.2 GIS Framework

ESRI ArcView is geographic information system (GIS) software for visualizing, managing, creating, and analyzing geographic data. The ESRI ArcObjects development toolkit was used to develop the prototype geospatial tools.

5.5.2.3 Progress

The progress in tool prototypes is summarized below. The following sections detail the prototyped tools further:

1. Find By Type Tools
    a. Find By Feature layer/class Tool - **Built**
    b. Find By Attribute Tool - **Built**
2. Find By Location Tool - **Built**
3. Find By Proximity Tools
    a. Simple Buffering Tool - **Built**
    b. Complex Buffering – Range Buffering Tool - **Built**
    c. Complex Buffering Tool
    d. Close Features Tool
4. Elevation Tools
    a. Line of Sight Tool - **Built**

b. ViewShed Tool

c. Slope Tool

5. Geometric Characteristics – Curvature Tool

6. Intersection Tool - **Built**

7. Routes Tool

8. Statistical Tool

9. Topology Tool

5.5.2.4 Filter By Attribute Tool

This tool Filters on Layer "LAP030" to find the roads with the attribute "RST = 2" in selected area (Blue rectangle). The message box shows the returned roads, by name, that match this filter. To use this tool, click on the "Find Roads by Attribute" tool. Draw a rectangle on the screen and the following will be displayed as long as there are matches in the selected area.

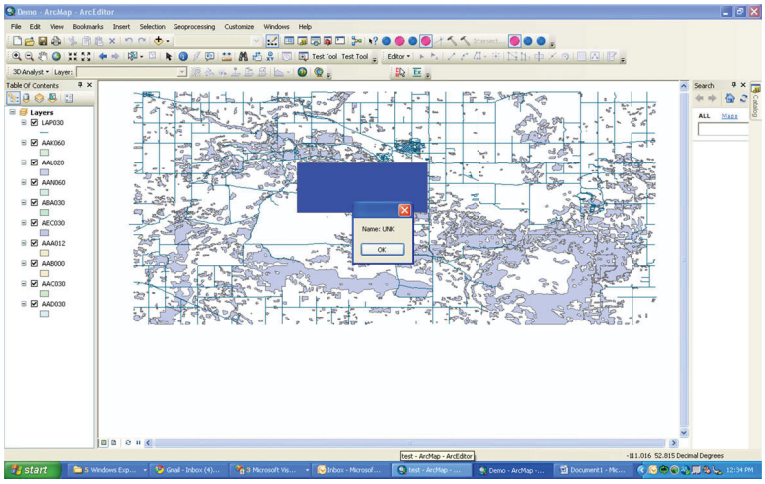

Figure 22: The filter by attribute tool.

5.5.2.5 Find By Type Tool

This tool finds all Layers that are of a particular type (polygon or line, etc.) in the map. This tool demonstrates that layers can be found by their name within the map. The message box shows the returned layers, by name, that match this filter. To use this tool click on the "Find by type" button and the following will be displayed as long as the layer (of that type) exists in the map.

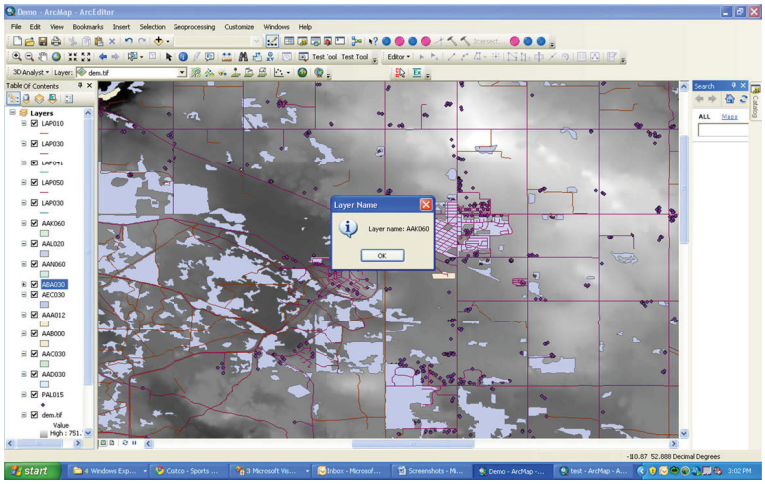

Figure 23: The find by type tool.

5.5.2.6 Select then Buffer

This tool creates a buffer around selected features on the map.

1. Select Features: Using the built in "Select by Rectangle" tool within ArcMap, select an area on the map that you would like to buffer the features within.

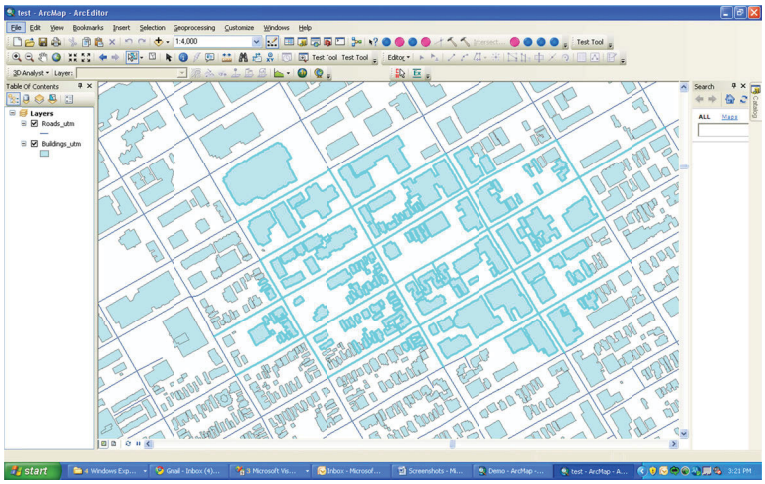

Figure 24: Selecting features in ArcMap

2. Buffer: Click on the Button to create buffer. There is now a buffer around all the features that were selected.

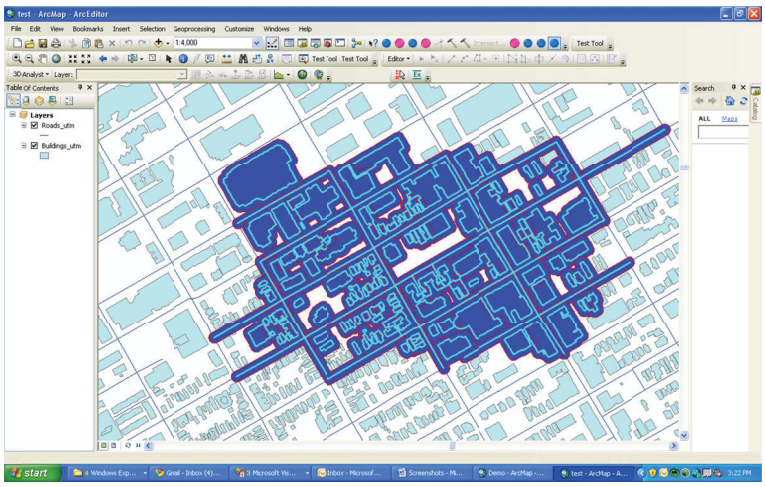

Figure 25: The simple buffer tool.

A second example of the tool with Wainwright data is outlined below.

1. Select Features: Using the built in “Select by Rectangle” tool within ArcMap, select an area on the map that you would like to buffer the features within.

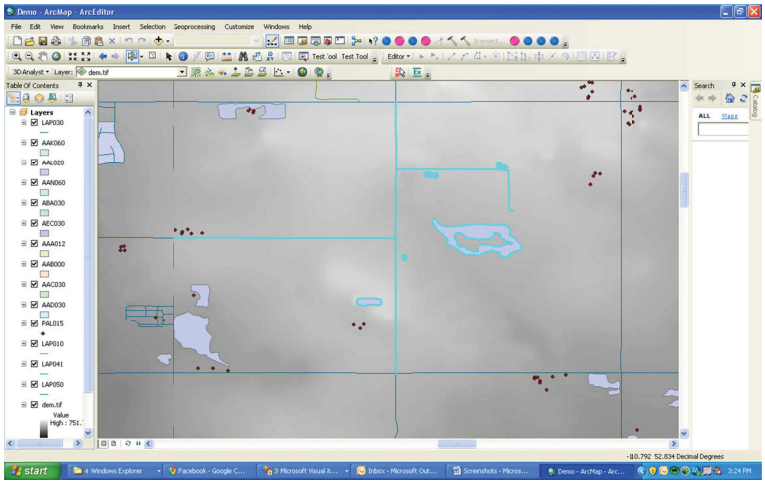

Figure 26: Feature selection in the Wainwright data.

2. Buffer: Click on the Button to create buffer. There is now a buffer around all the features that were selected.

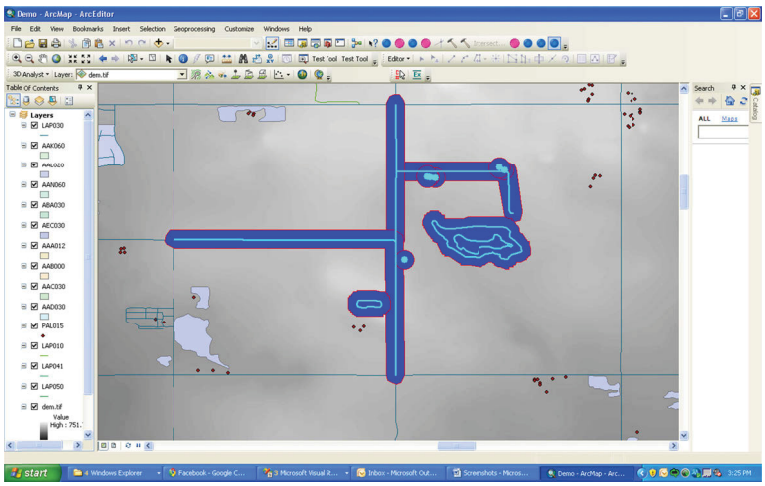

Figure 27: The buffering tool on Wainwright data.

5.5.2.7 Filter By Proximity Simple Tool

This test deletes all features around a point on the map. This tool indicates features that are within a certain distance around a "target location" for example. To use, click on the "Find by proximity simple" button.

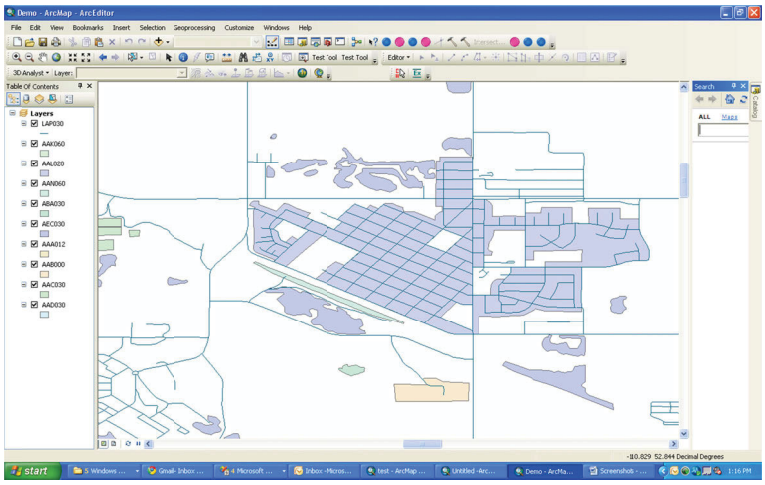

Figure 28: Before the filter by proximity test.

After the button is pressed, the building features around the point are now deleted.

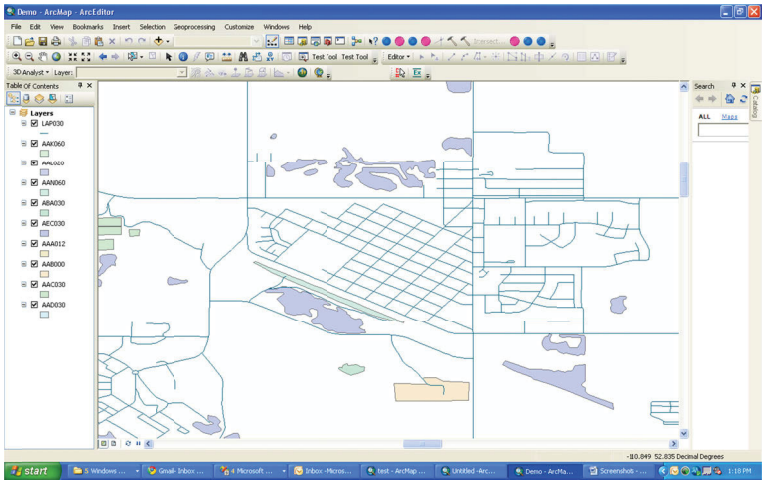

Figure 29: After the filter by proximity test.

5.5.2.8 Find By Proximity Complex

This test uses the find by proximity-complex tool to deletes all features around a point on the map that are within a range (e.g. 200m < Point < 800m). This tool indicates features that are within a certain range around a "target location" for example. To perform this test, click on the "Find by proximity complex" button.

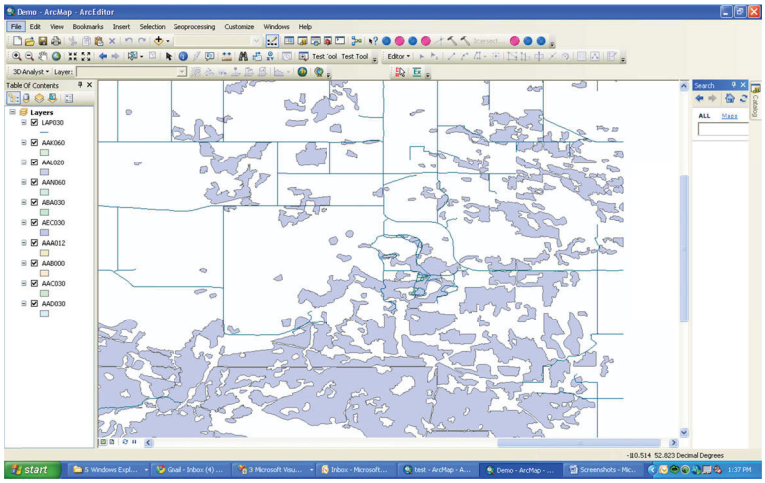

Figure 30: Before the find by proximity-complex test.

Click on the "Find by Proximity complex" button.

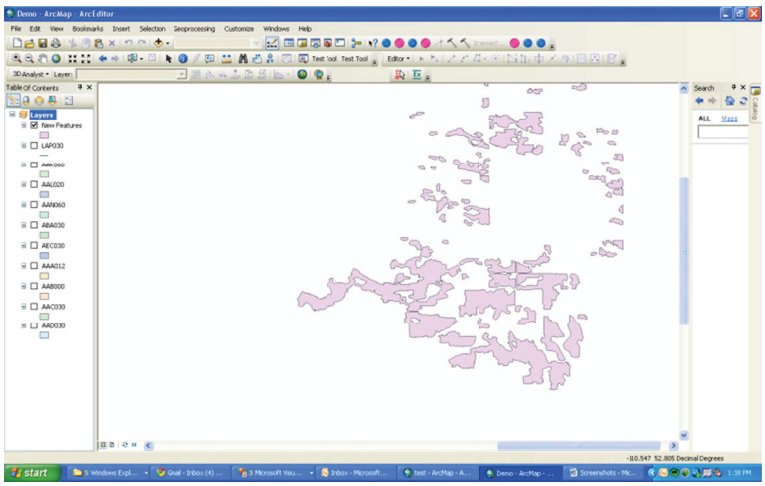

Figure 31: After the find by proximity-complex test.

Example 2, Before the tool is executed:

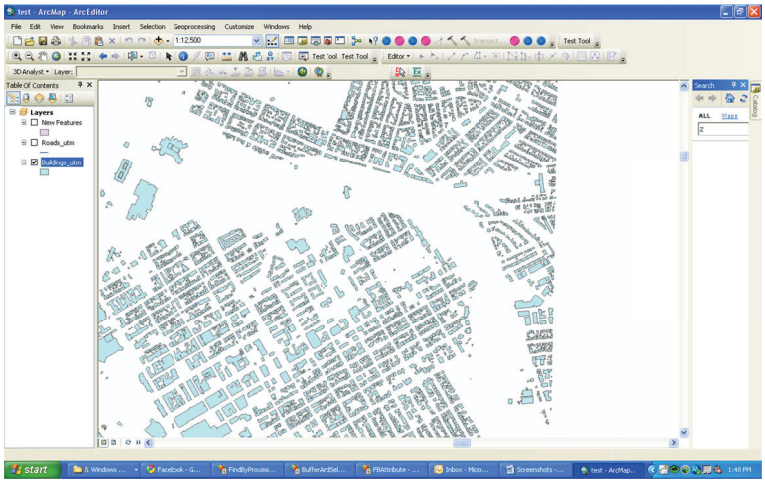

Figure 32: Source data for another example of the find by proximity-complex tool.

Click on the “Find by Proximity complex” button.

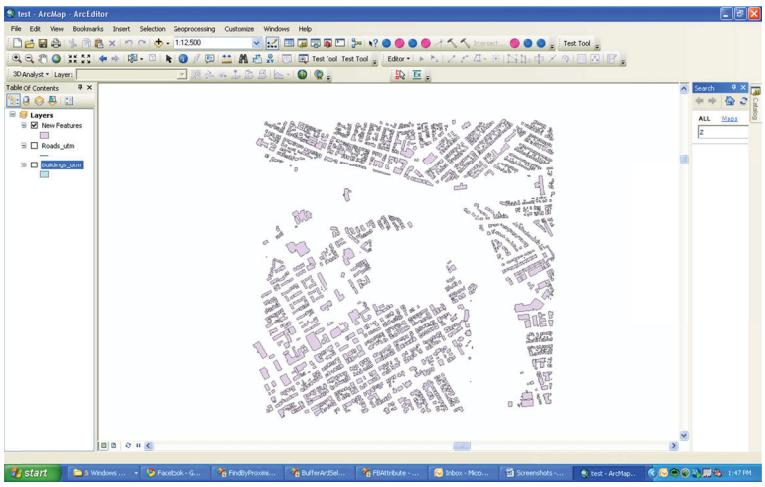

Figure 33: The results of another example of the find by proximity-complex test.

5.5.2.9 Geometric Intersection Tool

This tool will find the intersections of features on the map. In this example, the tool finds where roads that are buffered intersect with forested areas on the map. The tool can be used to fine where roads are canalized by buildings or other features.

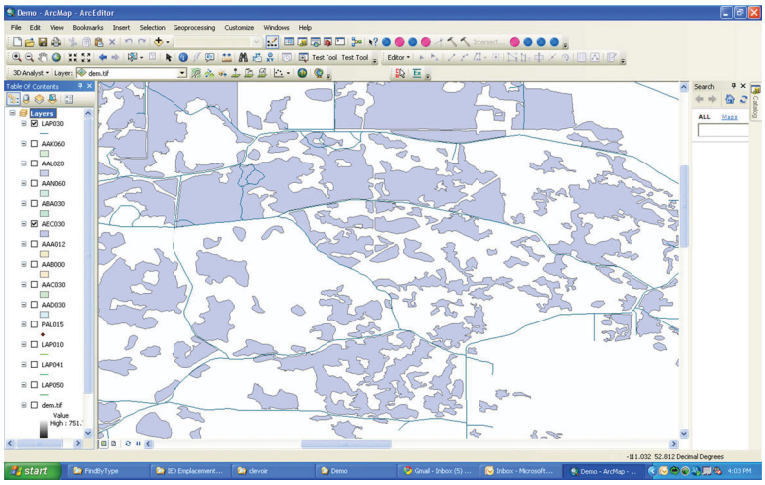

Figure 34: Road and forest data for Wainwright.

Use the buffer tool on the roads layer

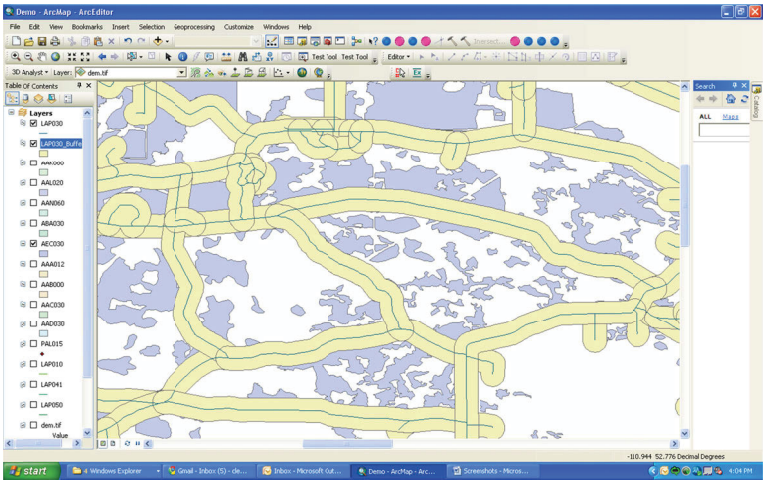

Figure 35: Buffered roads in Wainwright.

Use the intersect tool to find the intersections of the road layer and the forested area layer.

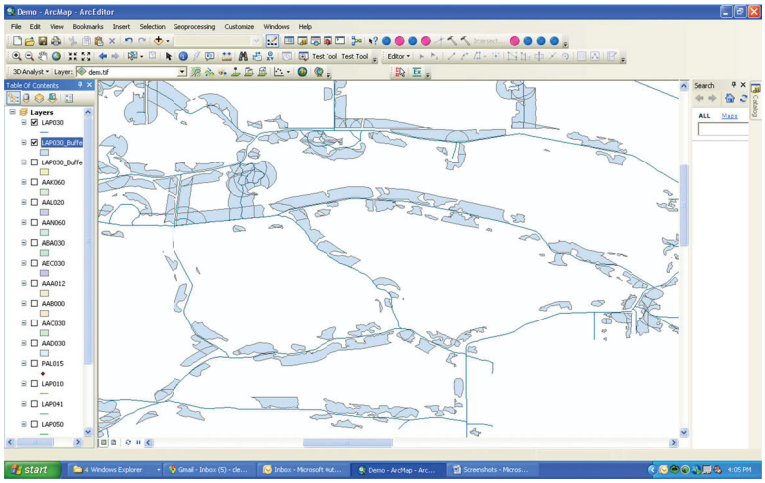

Figure 36: Intersection results showing all forest within the buffer distance of a road.

5.5.2.10 Line of Sight Tool

This tool indicates the line of sight from one point location to another point location on the map. The green areas indicated full line of sight on that path and the red indicates there is no line of sight on that path. This tool can be used to determine whether spotters have line of sight to a device location for example.

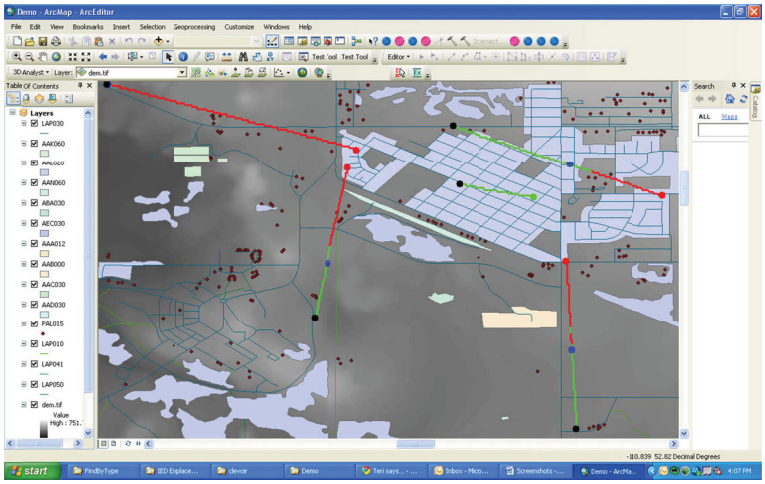

Figure 37: The line of sight tool.

6 Next steps

This section outlines some of the next steps recommended for the continuation of the work documented in this report.

6.1 User requirements definition

The work to date has built an excellent understanding of the technical requirements derived from the objective of creating plausible IED attack scenarios. However, at this stage, the user and the user need are only notionally defined. The CF training user community needs to be defined in detail and the detailed user requirements need to be developed through interviews and trials with members of the user community.

6.2 Iterative OMI Design, Development and Evaluation of Software

In this section we present a high-level overview of an iterative Human Factors Operator Machine Interface (OMI) design, development, and evaluation stream that should be planned to develop an IED scenario generation software tool. The detailed design effort for the software involves three design iterations; specifically (see Figure 38):

1. Iteration 1 – High-Level OMI Concepts

2. Iteration 2 – Preliminary Screen Design

3. Iteration 3 – Dynamic OMI Design and Development

Each design iteration will follow a standard Human Factors Engineering (HFE) process during which there will be design (i.e. develop ideas), prototype (i.e. make those ideas 'real', to some extent), and evaluate (i.e. present to SMEs). This process may 'spiral' through several cycles within one iteration as it is evaluated according to different perspectives and coalesces on the most appropriate design. The output from each step in the process will feed into the next step, and the output from each iteration will feed into the next iteration.

Three separate evaluation methodologies will be developed to suit the specific needs of each design iteration. However, all methodologies include the following stages:

1. **Subject Matter Expert (SME) Evaluation.** Consists of informal (i.e., less structured) group reviews with members of the project team (e.g. a small set of 'user advocates' ideally with operational experience) that would interact with the project team on a regular basis and representatives (i.e., SMEs) of the operational community to 'de-bug' the design before the more formal user validation sessions. A short cycle of iterations would be required (2 weeks to a month) to conduct an SME evaluation; and;

2. **Formal User Evaluation.** Consists of formal (i.e., structured) group reviews with representatives of the operational community to validate and iterate the design. The formal evaluation would be conducted using a task-based structure. These evaluations could include the development of a test

plan and formal documentation of the results. A long cycle of iterations would be required (2 to 4 months) to conduct a formal user evaluation.

The following sections summarize the evaluation methodologies underpinning each of the three evaluation iterations.

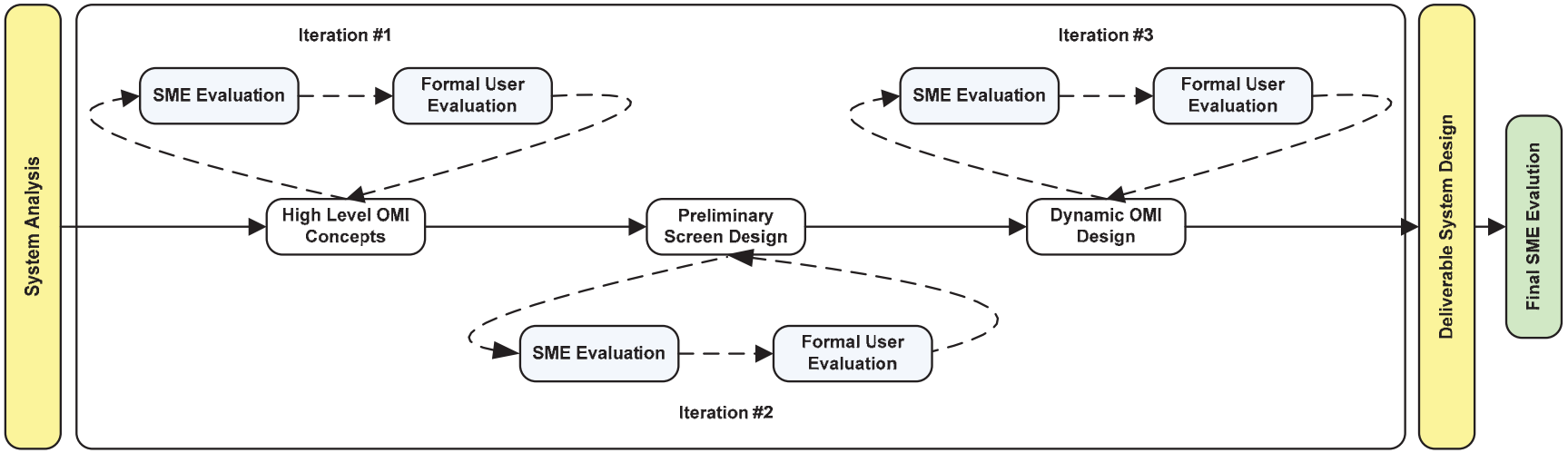

Figure 38: Human Factors OMI Evaluation Stream

6.2.1 Iteration 1 – High-Level OMI Concepts

The overall high-level concepts for the OMI will be developed in collaboration with the Scientific Authority and domain expert(s) from the user community. This will be followed by an evaluation session with representative users from the project team (SME Evaluation) and the wider operational community (Formal User Evaluation) to validate the high-level OMI concepts. Tasks in this phase include:

1. **Establish a high level structure for the OMI.** The high level structure of the suite of screens will be defined, along with the primary user (e.g. trainer) task flows through those screens based on interviews with members of the user community. The latter would be used as the basis for human engineering analysis and design. In addition, the high level menu hierarchy for the display suite will be roughly defined. This will include both hierarchy and screen navigation protocols. This effort will be coordinated to synchronize with each software build and will form the basis of the software 'presentation layer'.

2. **Develop primary information for the display protocol tasks.** The primary information to be presented on displays (functional, physical, administrative) and its high level organization as well as the primary controls for interacting with that information will be defined and conceptually designed. The outputs of the menu, screen, and control/display protocol tasks will be documented.

3. **Conduct evaluations to validate OMI design constructs and initial screen designs.** The HFE team will develop a visual presentation (i.e., low fidelity prototype) of the OMI concept defining the role of the user, the critical tasks, the initial conceptual design of the screens, and the high level user task flow through the system within the context of a mission scenario. The HFE team will use these materials to conduct a structured, task-based, user evaluation of the conceptual design. Issues raised during the validation sessions will be documented and will catalogue the disposition of the issues raised (Accepted into the Design, Rejected with Rationale, or Under Investigation by the team). The HFE team will update the screen designs to incorporate the results of the evaluation and extend the analysis in collaboration with the Scientific Authority.

6.2.2 Iteration 2 – Preliminary Screen Design

Based on the overall high-level concepts for the OMI as defined during Iteration 1, the preliminary screens will be designed and the associated operator interaction further refined. This will be followed by evaluation sessions with representative users from the project team (SME Evaluation) and the wider operational community (Formal User Evaluation) to validate the screen designs. Tasks in this phase include:

1. **Design and Prototype Preliminary Screens.** Preliminary screen designs will be defined to address functionality associated with a range of tasks. In addition, user task flows through those screens based on the critical tasks and the mission scenario will be further refined. The outputs of the menu, screen, and control/display protocol tasks will be documented.

2. **Evaluate OMI Design Constructs – Static Screen Review.** The HFE team will develop a visual presentation (i.e., low fidelity prototype) of the preliminary screens for the OMI and the user task flow through the system within the context of the mission scenario. The team could also develop user review questions to obtain structured information on perceived adequacy and usability of the concepts, and the perceived impact of this concept on user task performance and workload. The HFE team would use these materials to conduct a structured, goal-based, user review of the conceptual design. Issues raised during this evaluation could be documented and catalogued (Accepted into the Design, Rejected with Rationale, or Under Investigation by the team). The HFE team will update the screen designs to incorporate the results of the working group and extend the analysis in collaboration with the Scientific Authority to include all screens within the environment.

6.2.3 Iteration 3 – Dynamic OMI Design

Upon validation of the conceptual design and preliminary screen designs, the OMI will be prototyped. This will be followed by evaluation sessions with representative users from the project team (SME Evaluation) and the wider operational community (Formal User Evaluation) to validate the screen designs. Tasks in this phase include:

1. **Prototype OMI.** The complete set of preliminary screens will be prototyped that will illustrate the updated OMI, the updated user role, the updated user task flows, and a more complete preliminary screen design set to the user. This prototype will depict the screens from the perspective of the user task flow within the context of the composite mission scenario.

2. **Validate Screen Designs – Dynamic Screen Review.** At this stage, validation of the OMI design will be performed through structured usability testing. SMEs (e.g., C-IED Trainers) will complete a set of real tasks with the prototype (i.e. the prototype interface should be 'driven' by signals in order to seem to the users that there is actually a task scenario in process). The HFE Team could observe their performance and collect empirical data (e.g., errors made and difficulties experienced). Participants would be instructed to talk out loud while performing the tasks to help elucidate their decision making processes. The data collected can be applied to remedy the observed usability problems by going through OMI design iteration. The team could develop updated questionnaires to obtain more detailed user input on usability, task performance, and workload. Issues raised during this working group would be documented and would catalogue the disposition of the issues raised (Accepted into the Design, Rejected with Rationale, or Under Investigation by the team).

3. **Update Screen Designs.** The HFE team will update the screen designs and documentation to incorporate the results of the dynamic OMI design review.

6.3 Cognitive Task Analysis development

The Cognitive Task Analysis (CTA) developed in this task provides a comprehensive decomposition of documented activity related to IED threat activity. Further effort should be made to define the information requirements associated with task performance in the CTA, and to

provide further traceability from the CTA to the software requirements. In parallel with the OMI development iterations defined above, SME review should be used to validate the CTA.

6.4 Coordination with DLSE

As the overall objective is the training capability, not the scenarios in and of themselves, coordination with DLSE will be important in the next phase of development. As VBS2 is the platform for training delivery, care must be taken to focus the scenario generation development on aspects of scenarios that carry over into the current or planned training capability with VBS2.

Coordination will be required to define the scope and nature of the information required for scenario export to VBS2.

6.5 Define information requirements

The analysis to date has produced a body of knowledge on the types of spatial features important in IED attacks. The knowledge captured also defines the ways in which these features relate to IED tactics or blue force considerations. This information can be used to define the traceability between spatial data content and resolution, and IED tactic components. This traceability, in turn, can be used to either, a) define the information requirements necessary to model all IED tactics, or b) the range of tactics that can be created plausibly in data of a certain spec.

The figure below shows representative data from Wainwright, developed to the Vector MAP (VMAP) level 2 specification. This data is comprehensive in feature classification and attribution, and will support a range of the courser spatial characteristics of IED operations. However, it is lacking in spatial resolution and feature detail to support some aspects of IED tactics, such as the identification of canalized road sections.

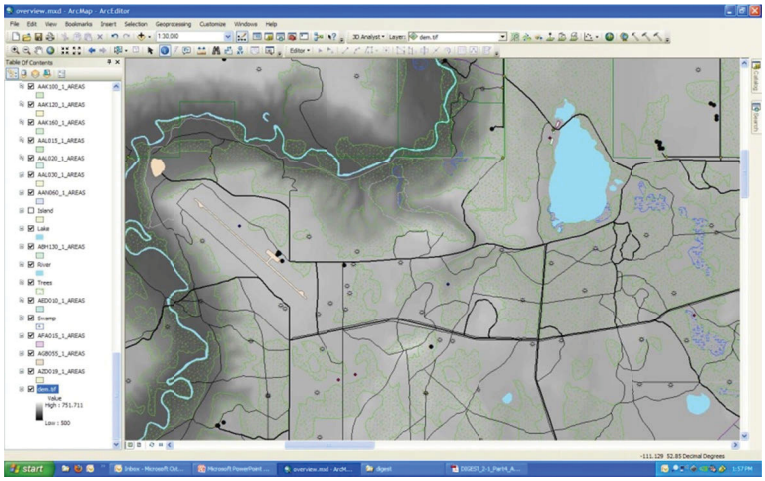

Figure 39: Representative data from Wainwright.

The data in the figure below was preserved from the generation of a high resolution, simulation terrain database. This data shows many finer characteristics, such as the exact widths of road features, and the existence of many potentially canalizing features, such as buildings and

embankments. This data will support a wide range of the queries suggested by the analysis of insurgent tactics, but requires the export of data from the terrain generation process. This should be possible in the future production of terrain databases, but may not be possible retroactively for databases that have already been produced.

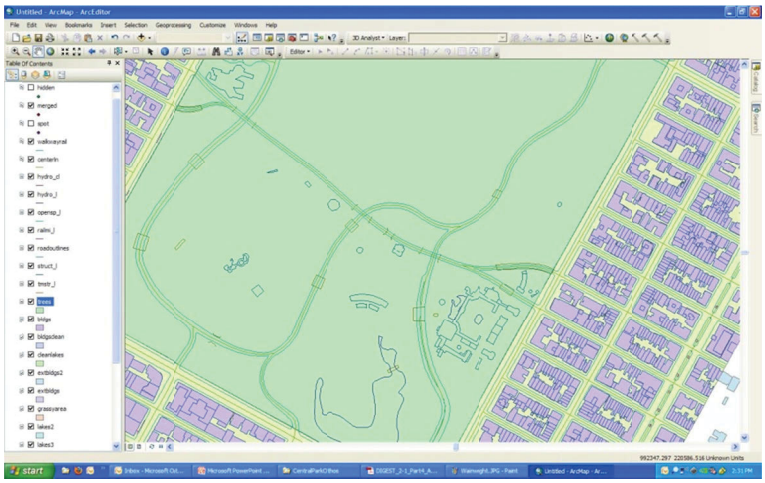

Figure 40: Representative data from a simulation terrain development project.

6.6 Lessons learned

Significant progress has been made to date. Information has been captured on typical IED attack scenarios. Analysis has been performed of the general technical requirements arising from typical IED attack scenarios. This led to the development of an architecture for a scenario generation tool. A key component of this architecture is the identification of knowledge management and geospatial processing requirements. Initial prototyping activities have produced the initial elements of knowledge processing and geospatial processing engines.

In the prototyping process, a number of observations where made:

1. Protégé/Fact+ is sufficient for prototyping, but is not stable enough for production. Alternate tools, such as SWI-Prolog (with performs a more constrained subset of reasoning) or RacerPro (which will incur a run-time licensing cost on the scenario generation tool) should be researched.

2. ArcObjects has sufficient capability to be the core of the geospatial processing engine. However, the implementation complexity for the use of ArcObjects is higher than initially estimated. Some tools, such as view shed and route analysis, may have to be developed outside of ArcObjects.

The knowledge base modifications to update tactical knowledge will be a potentially complex task. Attention will have to focus on the development of the user requirements and work flow for this task.

References

[1] Army Lessons Learned Center (2009). LSR 09-019 Consolidated C-IED Summary Report.

[2] Army Lessons Learned Center (2010). LSR 10-003. C-IED Best Practices and Defensive TTP.

[3] Army Lessons Learned Center (2010a). LSR 10-006. Consolidated IED Report, May.

[4] Army Lessons Learned Centre (2010b). Counter-Improvised Explosive Devices. Dispatches: Lessons Learned for Soldiers (Vol. 15, No 1).

[5] Army Lessons Learned Center (2011). ILR Synopsis – Radio Controlled Improvised Explosive Device (RCIED) Attacks on Coalition Patrol Convoys.

[6] Defence Research and Development Canada (2010). Statement of work: automation of threat emplacement for training scenarios in synthetic environments.

[7] Department of National Defence (2010). Canadian Forces Counter Improvised Explosive Devices Tactics, Techniques and Procedures – Version 2. B-GL-356-022/FP-001.

[8] Department of National Defence (2011). Attack the Network Field Guide – Afghanistan. Draft – Working Document.

[9] Digital Geographic Information Working Group (2000). The Digital Geographic Information Standard Part 4, Feature and Attribute Coding Catalogue, Edition 2.1. Retrieved 03 23, 2011 from https://www.dgiwg.org/digest/DownloadDigest.htm.

[10] Eles, P. T. (2010). Characterizing the IED Threat: A Classification of IED events in Kandahar Province (Initial Results). DRDC CORA, CEFCOM Operational Research and Analysis Team.

[11] Jarmasz, J.& Eles, P. (2009). Notes. Patterns of IED Scenarios and Indicators. DRDC.

[12] MSDL Product Development Group. (2008). Military Scenario Definition Language (MSDL) - SISO-STD-007-2008. Retrieved from http://www.sisostds.org/DigitalLibrary.aspx?Command=Core_Download&EntryId=30830.

[13] Unrau, D. (2011). IED_knowledge.owl. OWL Ontology delivered to DRDC.

[14] Zobarich, R. (2011). Red Forces – Conduct an IED attack 28-march-2011.ta1. Task Architect file delivered to DRDC.

This page intentionally left blank.

List of symbols/abbreviations/acronyms/initialisms

ALLC	Army Lessons Learned Centre
CF	Canadian Forces
C-IED	Counter Improvised Explosive Device
CTA	Cognitive Task Analysis
DIGEST	Digital Geographic Information Exchange Standard
DND	The Department of National Defence
DRDC	Defence Research and Development Canada
FACC	Feature and Attribute Coding Catalogue
GIS	Geographic Information System
HBR	Human Behaviour Representation
IED	Improvised Explosive Device
LSEC	Land Software Engineering Centre
LSEC	Land Software Engineering Centre
MSDL	Military Scenario Definition Language
OWL	Web Ontology Language
R&D	Research and Development
RCIED	Radio Controlled Improvised Explosive Device
RDF	Resource Description Framework
SME	Subject Matter Experts
TDP	Technology Development Project
UI	User Interface
VBS2	Virtual Battle Space 2
VMAP	Vector Map

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (The name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.)

Publishing: DRDC Toronto
Performing: CAE PS Canada, 1135 Innovation Drive Ottawa, ON K2K 3G7
Monitoring:
Contracting: DRDC Toronto

2. SECURITY CLASSIFICATION (Overall security classification of the document including special warning terms if applicable.)

UNCLASSIFIED
(NON-CONTROLLED GOODS)
DMC A
REVIEW: GCEC JUNE 2010

3. TITLE (The complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S, C or U) in parentheses after the title.)

Automation of IED Threat Emplacement for Training Scenarios

4. AUTHORS (last name, followed by initials – ranks, titles, etc. not to be used)

Unrau, D.; Zobarich, R.; Levoir, C.

5. DATE OF PUBLICATION (Month and year of publication of document.)	6a. NO. OF PAGES (Total containing information, including Annexes, Appendices, etc.)	6b. NO. OF REFS (Total cited in document.)
October 2011	73	14

7. DESCRIPTIVE NOTES (The category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Contract Report

8. SPONSORING ACTIVITY (The name of the department project office or laboratory sponsoring the research and development – include address.)

Defence R&D Canada – Toronto
1133 Sheppard Avenue West
P.O. Box 2000
Toronto, Ontario M3M 3B9

9a. PROJECT OR GRANT NO. (If appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant.)	9b. CONTRACT NO. (If appropriate, the applicable number under which the document was written.)
12RR03	W771-06-8100-14

10a. ORIGINATOR'S DOCUMENT NUMBER (The official document number by which the document is identified by the originating activity. This number must be unique to this document.)	10b. OTHER DOCUMENT NO(s). (Any other numbers which may be assigned this document either by the originator or by the sponsor.)
DRDC Toronto CR 2011-134	5180-001 ver 04

11. DOCUMENT AVAILABILITY (Any limitations on further dissemination of the document, other than those imposed by security classification.)

Unlimited distribution

12. DOCUMENT ANNOUNCEMENT (Any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in (11) is possible, a wider announcement audience may be selected.))

Unlimited distribution

13. ABSTRACT (A brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual.)

(U) In contemporary operations, asymmetric threats, especially Improvised Explosive Devices (IEDs), are a leading cause of Canadian Forces casualties and injuries. Automation to support the development of better threat scenarios for training exercises would improve the Canadian Forces' capability to prepare soldiers for future missions involving emerging threats. This report summarizes progress to date on the design of a software tool to automate the generation of plausible IED threat scenarios. The core components of the design are a knowledge processing engine and a geospatial processing engine. The knowledge processing engine will act on a database (ontology) of insurgent tactics to translate user−supplied constrains and training objectives into threat scenario characteristics. The geospatial processing engine will query map data to determine locations for scenario components, such as the device location and spotter locations. Initial technical prototyping has demonstrated the feasibility of this approach. An iterative operator−machine interface development process, cycling through design, prototyping and evaluation phases is suggested as the next step in development.

(U) Dans les opérations contemporaines, les menaces asymétriques, en particulier les dispositifs explosifs de circonstance (IED), sont une cause majeure de blessures et de décès chez le personnel des Forces canadiennes. L'automatisation servant à appuyer le développement de meilleurs scénarios de menace pour les exercices de formation améliorerait la capacité des Forces canadiennes à préparer les soldats pour des missions à venir mettant en cause de nouvelles menaces. Ce rapport résume le progrès realise jusqu'à maintenant sur la conception d'un outil logiciel servant à automatiser la production de scénarios de menace d'IED plausibles. Les éléments essentiels du concept sont un moteur de traitement des connaissances et un moteur de traitement géospatial. Le moteur de traitement des connaissances agira sur une base de données (ontologie) de tactiques des insurgés pour traduire des objectifs de formation et des contraintes fournies par l'utilisateur en caractéristiques de scénario de menaces. Le moteur de traitement géospatial interroge les données cartographiques pour déterminer les emplacements pour les éléments scénario, comme l'emplacement du dispositif et les emplacements des guetteurs. Le prototypage technique initial a démontré la faisabilité de cette approche. Un processus de développement d'interface opérateur machine de type itératif, parcourant les phases de conception, de prototypage et d'évaluation est suggéré comme prochaine étape en développement.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (Technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

IEDs; Counter−IED; training; simulation−based training; GIS; ArcGIS; scenario generation; Virtual Battle Space 2; VBS2; automatic scenario generation

Defence R&D Canada

Canada's Leader in Defence
and National Security
Science and Technology

R & D pour la défense Canada

Chef de file au Canada en matière
de science et de technologie pour
la défense et la sécurité nationale

DEFENCE R&D DÉFENSE

www.drdc-rddc.gc.ca